

An Inertial Dual-State State Estimator for Precision Planetary Landing with Hazard Detection and Avoidance

Robert H. Bishop*
The University of South Florida

Timothy P. Crain,[†] Chad Hanak[†]
Intuitive Machines, Inc.

Kyle DeMars[‡]
Missouri University of Science and Technology

John M. Carson III[§]
NASA Johnson Space Center

Nikolas Trawny[¶]
NASA Jet Propulsion Laboratory

John Christian^{||}
West Virginia University

The navigation filter architecture successfully deployed on the Morpheus flight vehicle is presented. The filter was developed as a key element of the NASA Autonomous Landing and Hazard Avoidance Technology (ALHAT) project and over the course of 15 free flights was integrated into the Morpheus vehicle, operations, and flight control loop. Flight testing completed by demonstrating autonomous hazard detection and avoidance, integration of an altimeter, surface relative velocity (velocimeter) and hazard relative navigation (HRN) measurements into the onboard dual-state inertial estimator Kalman filter software, and landing within 2 meters of the vertical testbed GPS-based navigation solution at the safe landing site target. Morpheus followed a trajectory that included an ascent phase followed by a partial descent-to-landing, although the proposed filter architecture is applicable to more general planetary precision entry, descent, and landings. The main new contribution is the incorporation of a sophisticated hazard relative navigation sensor—originally intended to locate safe landing sites—into the navigation system and employed as a navigation sensor. The formulation of a dual-state inertial extended Kalman filter was designed to address the precision planetary landing problem when viewed as a rendezvous problem with an intended landing site. For the required precision navigation system that is capable of navigating along a descent-to-landing trajectory to a precise landing, the impact of attitude errors on the translational state estimation are included in a fully integrated navigation structure in which translation state estimation is combined with attitude state estimation. The map tie errors are estimated as part of the process, thereby creating a dual-state filter implementation. Also, the filter is implemented using inertial states rather than states relative to the target. External measurements include altimeter, velocimeter, star camera, terrain relative navigation sensor, and a hazard relative navigation sensor providing information regarding hazards on a map generated on-the-fly.

I. Introduction

The extended Kalman filter has long been the workhorse of space-based navigation systems for on-orbit rendezvous, ranging from the early days of Apollo to the more recent era of the Space Shuttle. The Kalman filter is well-known and well-documented in the literature, dating from Kalman's original work in 1960.¹ We pose the precision planetary landing problem as a rendezvous problem with an intended landing site. For the required precision navigation system that is capable of navigating along a descent-to-landing trajectory to a precise landing, the impact of attitude errors on the translational state estimation are included in a fully integrated navigation structure in which

*Department of Electrical Engineering, 4202 E. Fowler Ave., Tampa, FL 33620, AIAA Fellow

[†]3700 Bay Area Blvd, Pasadena, TX 77058, AIAA Senior Member

[‡]Mechanical & Aerospace Engineering, 400 W. 13th Street, Rolla, MO 65409, AIAA Member

[§]Aeroscience & Flight Mechanics Division (IPA Detail from JPL), Houston, TX 77058, AIAA Associate Fellow

[¶]NASA Jet Propulsion Laboratory, Pasadena, CA, AIAA Member

^{||}Mechanical & Aerospace Engineering, Morgantown, WV 26506, AIAA Member

translation state estimation is combined with attitude state estimation. In addition, the map tie errors are estimated as part of the process, thereby creating a dual-state filter implementation. Also, the filter is implemented using inertial states rather than states relative to the target.

The problem that motivates our filter architecture approach is planetary descent-to-landing, however, the application of the proposed dual-state inertial extended Kalman filter is not limited to this class of problems. For example, this filter architecture was deployed on the Morpheus flight vehicle that followed a trajectory that included an ascent phase followed by a partial descent-to-landing. As the trajectory progresses, external measurements may include altitude (i.e., altimeter), relative velocity (e.g., velocimeter), attitude (e.g., star camera), position relative to a pre-loaded map (e.g., a terrain relative navigation (TRN) sensor), and a hazard relative navigation (HRN) sensor providing information regarding hazards to the targeting subsystem, as well as measurements relative to a map generated on-the-fly. An internal inertial measurement unit (IMU) provides measurements of non-gravitational accelerations and body rates. The location of the ground features are assumed to be corrupted by map-tie errors, all external sensors and the IMU are assumed to produce measurements exhibiting systematic and random errors, estimates of the initial position and velocity of the spacecraft are known with initial estimation errors assumed zero-mean with given state estimation error covariance, and uncertainty in the environment models (namely, the gravity) is assumed.

Navigation algorithms based on extended Kalman filter architectures are model-based. Since this paper focuses on the navigation algorithms themselves, our attention is on the environment and sensor models employed in the EKF and not on the high-fidelity models that might be deployed in the various numerical simulations to model the true state history of the spacecraft and to generate the simulated measurements. The models of the translational and rotational dynamics of the lander within the EKF are key to accurately predicting the motion of the spacecraft. Furthermore, without an accurate measure of the non-gravitational acceleration and the spacecraft angular velocity as provided by an accelerometer/gyro package, the translational and rotational dynamics models would not be capable of supporting precision landing. The navigation algorithm must also possess accelerometer and gyro models. Also of interest in precision landing is the location of the landing site. In performing precision descent-to-landing navigation, it is necessary to be able to accurately predict the location of the landing site, accounting for the presence of map tie errors. In addition to the dynamical models describing translation and rotation of the lander, it is necessary to formulate the dynamics of the estimation errors. The estimation errors represent the differences between the true spacecraft state (position, velocity, and attitude) and the navigation provided estimated state. Other error dynamics associated with systematic errors are also included in the development.

A. Organization of the Paper

The organization of this paper is as follows: The descent-to-landing scenario and associated reference frames are presented in Section II. This is followed in Section III by an overview of the basic structure of the extended Kalman filter (EKF). The necessary dynamics modeling, including a description of the nonlinear equations, a treatment of the linearized error equations, and a description of the modeling of accelerometers, gyros, and a generic landing site are presented in Section IV. The proposed EKF sensor models are presented in Section VI. The models that are implemented in the filtering algorithms are presented in Section VII along with some notes on implementation issues, such as IMU thresholding and residual editing. A summary of the recommendations for preparation for implementation in the Morpheus lander is presented along with a description of the filter implementation as it was realized for the Morpheus flights is given in Section ??.

B. Mathematical Notation

The following notation is used throughout for variables:

- Scalar quantities are given by non-bold lowercase and uppercase symbols, such as: a , A , γ , and Γ .
- Vector quantities are given by bold lowercase symbols, such as: \mathbf{a} and $\boldsymbol{\gamma}$.

- Matrix quantities are given by bold uppercase symbols, such as: \mathbf{A} and $\mathbf{\Gamma}$.
- Quaternions are given by bold lowercase symbols with overbars. Furthermore, the vector part of the quaternion is given by the same symbol in bold with no overbar and the scalar part of the quaternion is given by the same symbol in non-bold with no overbar. The magnitude of the quaternion is given the same symbol as the quaternion, but non-bold and retaining the overbar.

$$\bar{\mathbf{q}} = \begin{bmatrix} \mathbf{q} \\ q \end{bmatrix} \quad \text{and} \quad \bar{q} = \|\bar{\mathbf{q}}\| \quad (1)$$

The expected value of a variable is denoted by $E\{\cdot\}$. The vector or matrix two-norm is denoted by $\|\cdot\|$. The set of all real numbers is represented by \mathbb{R} , such that an n -dimensional space of real numbers is \mathbb{R}^n and a matrix which has n -rows and m -columns of real numbers is said to be in $\mathbb{R}^{n \times m}$.⁹ The Dirac delta is represented as $\delta(t - \tau)$ and is defined via the integral

$$\int_{-\infty}^{\infty} f(t)\delta(t - \tau)dt = f(\tau), \quad (2)$$

for any real-valued function $f(t)$ that is continuous at $t = \tau$. It is noted that the Dirac delta is zero for all $t \neq \tau$. The somewhat similar Kronecker delta is represented as δ_{ij} , and is defined such that

$$\delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}. \quad (3)$$

The transpose is denoted with a superscript “T” and the inverse by a superscript “−1.” Furthermore, the superscripts “−” and “+” represent *a priori* and *a posteriori* values (i.e. values which immediately precede and succeed a measurement update). A quantity which is expressed in a particular frame is given a lowercase symbol in the superscript to denote in which it is represented. The hat accent ($\hat{\cdot}$) is used to denote an estimated quantity, and the dot accent ($\dot{\cdot}$) is used to denote a temporal derivative. The inner (or dot) product is denoted by \odot . The cross product matrix, denoted by $[\cdot \times]$, is such that $\mathbf{a} \times \mathbf{b} = [\mathbf{a} \times] \mathbf{b}$, where $\mathbf{a}, \mathbf{b} \in \mathbb{R}^3$. In terms of the components of \mathbf{a} , the cross product matrix, $[\mathbf{a} \times]$, is

$$[\mathbf{a} \times] = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}. \quad (4)$$

The purely off-diagonal symmetric matrix, $[\mathbf{a} \times \mid]$ and the diagonal matrix $[\mathbf{a} \setminus]$ are given by

$$[\mathbf{a} \times \mid] = \begin{bmatrix} 0 & a_3 & a_2 \\ a_3 & 0 & a_1 \\ a_2 & a_1 & 0 \end{bmatrix} \quad \text{and} \quad [\mathbf{a} \setminus] = \begin{bmatrix} a_1 & 0 & 0 \\ 0 & a_2 & 0 \\ 0 & 0 & a_3 \end{bmatrix}, \quad (5)$$

respectively. Furthermore, note that for any $\mathbf{a}, \mathbf{b} \in \mathbb{R}^3$ the following properties hold

$$[\mathbf{a} \times] \mathbf{b} = -[\mathbf{b} \times] \mathbf{a}, \quad [\mathbf{a} \times \mid] \mathbf{b} = [\mathbf{b} \times \mid] \mathbf{a}, \quad \text{and} \quad [\mathbf{a} \setminus] \mathbf{b} = [\mathbf{b} \setminus] \mathbf{a}. \quad (6)$$

Given the two reference frames, denoted by the a -frame and the b -frame, the transformation matrix which maps vectors from the a -frame to the b -frame (or equivalently expresses the relative attitude of the a -frame with respect to the b -frame) is given as \mathbf{T}_a^b . Equivalently, the quaternion which relates the a -frame to the b -frame is given as $\bar{\mathbf{q}}_a^b$.

II. Descent-to-Landing Scenario and Reference Frames

A. Descent-to-Landing Scenario

A schematic representation of the descent-to-landing scenario is shown in Fig. 1 that illustrates the thrust-coast-thrust engine configuration, the nominal and map tie corrupted landing site and

surface features, a measurement of spacecraft attitude, and a measurement of position via processing of an image of the planet surface. The descent-to-landing begins in a parking orbit about the planet or on a direct entry trajectory. The descent trajectory is a thrust-coast-thrust trajectory. The first thrust phase is a de-orbit maneuver designed to lower the orbit periapsis where the landing sequence is initiated. Once the descent has been initiated, the spacecraft enters a coast phase until it reaches the orbit periapsis. After the coast phase, the braking burn occurs near periapsis and the final descent maneuver nulls the remaining vehicle velocity to achieve a soft landing. As the trajectory progresses, measurements of altitude, relative velocity, attitude (from a star camera), and position (from a terrain camera, which is subject to map tie error) are processed. A schematic representation of the descent-to-landing scenario, shown in Fig. 1, illustrates the thrust-coast-thrust engine configuration, the nominal and map tie corrupted landing site and surface features, a measurement of spacecraft attitude, and a measurement of position via processing of an image of the planet surface.

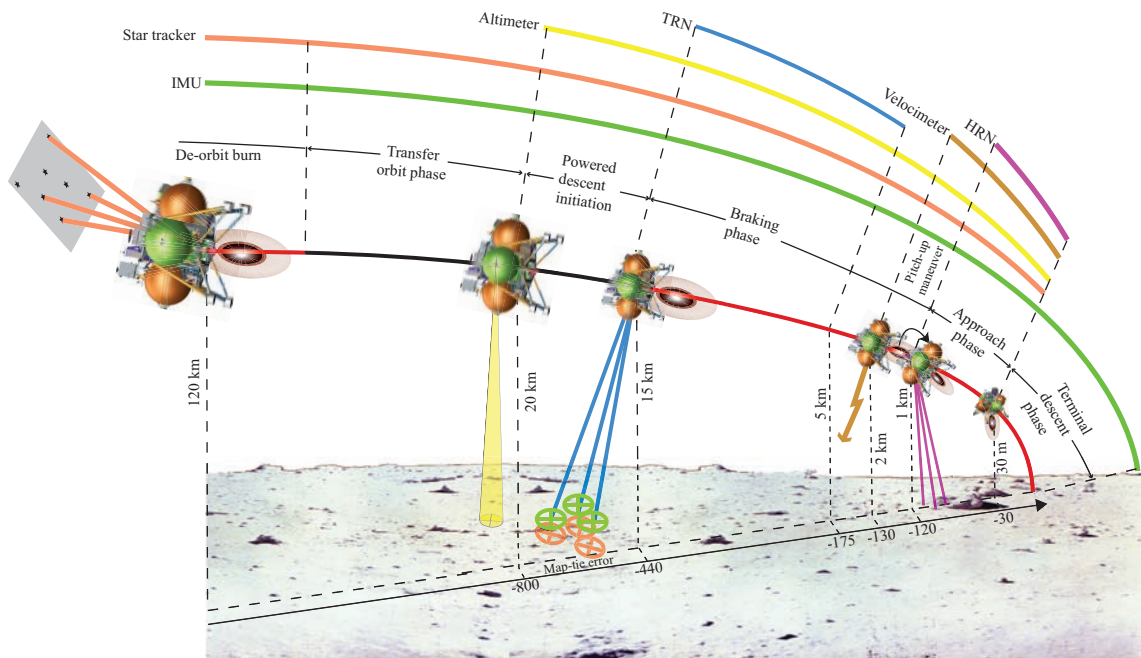


Figure 1. Schematic Descent-to-Landing Scenario

B. Reference Frames

Five basic classes of reference frames are considered: a planet-centered inertial frame, a planet-centered, planet-fixed reference frame, a planet-surface, planet-fixed reference frame, a spacecraft body reference frame, and various spacecraft sensor reference frames.

1. Planet-Centered Inertial Reference Frame

The inertial frame (denoted by i) is taken to be the J2000 reference frame centered at the celestial body (at the center of mass of the body). The J2000 reference frame is defined via the FK5 star catalog with a standard epoch defined as 1.5 January 2000, or 12 P.M. (noon) on January 1st 2000 in the Barycentric Dynamical Time (TDB) time scale.² The TDB time scale is a form of atomic time which allows Newtonian physics, corrected for general relativity, to be followed. The J2000 frame will serve as the base frame to which all other frames are related, which is consistent with the PDS Standards Reference.³ The J2000 frame is known as the planet centered inertial (PCI) frame.

The stellar reference frame (denoted by sr) is the frame in which the right ascension and declination of the set of stars is written. Typically, stellar reference frame is not aligned with

the J2000 reference frame, but is an inertial reference frame. The difference between the stellar reference frame and the J2000 inertial reference frame is a known constant transformation.

2. Planet-Centered, Planet-Fixed Reference Frame

There are two methods that we can utilize to compute the transformation from the planet-centered, planet-fixed (PCPF) reference frame (denoted by f) to the PCI reference frame, denoted by \mathbf{T}_i^f , or equivalently by the quaternion $\bar{\mathbf{q}}_i^f$. The first method implements the mathematical model provided by the IAU/IAG/COSPAR Working Group on Cartographic Coordinates and Rotational Elements of the Planets and Satellites² within the EKF algorithm. The north pole of the body is defined to be the pole of rotation which lies on the north side of the invariable plane of the solar system and is specified by its right ascension and its declination. The remaining required angle specifies the ephemeris position of the prime meridian. The mathematical models of these angles are specified in Archinal, et al.² To validate the implementation, the implemented PCPF frame can be compared to the same frame as implemented in the SPICE¹⁴ system. For example, our analysis of the lunar-center, lunar-fixed (LCLF) reference frame accuracy relative to the SPICE computation showed that error angle between the frames is found to be on the order of XX arc sec and that unit directions of the inertial axes at the surface of the Moon resulted in maximum norm of the position differences of less than $6\mu\text{m}$.

The second method uses an *i-load* initial transformation of the PCPF reference frame to the PCI reference frame at a given epoch (usually the start of the simulation or the initiation of the descent-to-landing trajectory) and then continuously updates the transformation employing a single rotation about the planet spin axis. This represents the simplest model for implementation in the EKF. In both cases, we assume that the transformation, $\mathbf{T}_i^f(t)$, is known exactly—it is not a source of uncertainty.

3. Planet Surface Fixed Reference Frame

The planet surface fixed reference frame (PSF) is a reference frame (denoted by s) fixed to a point on the planet surface. This topocentric reference frame is selected such that the unit directions of the frame point in the east, north, and zenith directions. The frame is fully defined by the spherical latitude and longitude of the point at which the frame is attached to the surface. Since the latitude and longitude of the point are defined in a fixed reference system, the PSF reference frame is defined relative to the PCPF reference frame, denoted by \mathbf{T}_f^s .

4. Spacecraft Body and Sensor Reference Frames

The spacecraft body frame (denoted by b) and the spacecraft sensor frames are all frames which are fixed to and unmoving with respect to the spacecraft. The body frame is a frame which is fixed along axes of the structure of the spacecraft. The spacecraft body frame is depicted in Fig. 2 by the unit vector triad $\{\mathbf{x}_b, \mathbf{y}_b, \mathbf{z}_b\}$.

The spacecraft sensor frames are centered at each corresponding sensor and are allowed independent alignments. Each sensor frame is defined by a unit vector triad, as shown in Fig. 2. Typically, the sensor frame $+\mathbf{z}$ -axis is defined to be along the bore of the sensor and the remaining axes span the plane perpendicular to the bore direction. Since both the spacecraft body frame and the spacecraft sensor frames are fixed to and unmoving with respect to the body, it is assumed that the relative orientation of each sensor frame with respect to the body frame is known via calibration and testing. The sensor reference frames considered include the IMU case reference frame denoted by c , the velocimeter frame denoted by v , the altimeter reference frame denoted by a , the star camera reference frame denoted by sc , the terrain relative navigation reference frame denoted by t , and the hazard relative navigation reference frame denoted by h . Onboard Morpheus, the sensor suite did not include a star camera or a terrain relative navigation sensor, hence these are not illustrated in Fig. 2. A summary of the designations of the reference frames considered is given in Table 1.

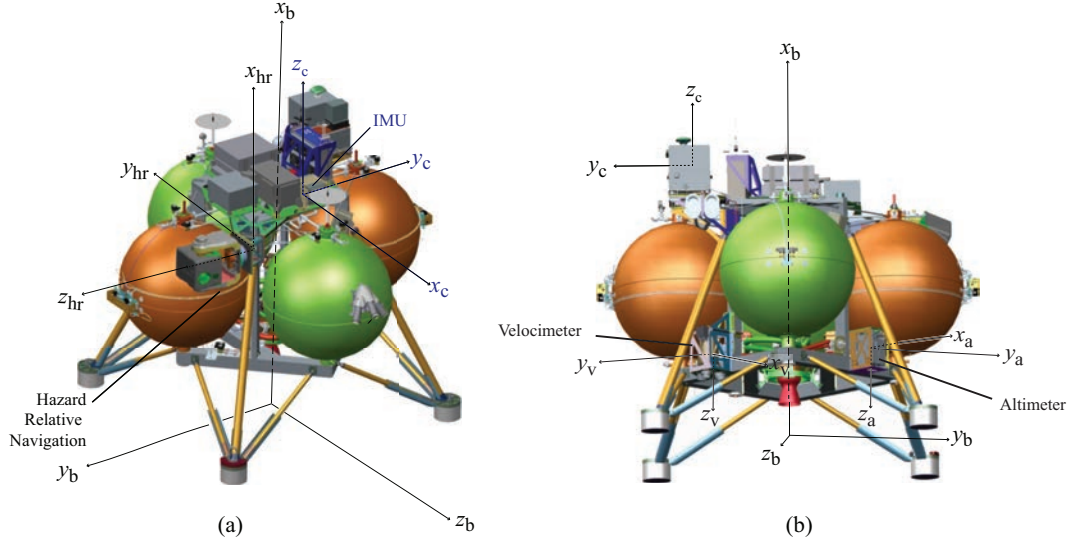


Figure 2. Spacecraft Body Fixed and Sensor Fixed Reference Frames. (a) front view, (b) rear view.

Table 1. Reference Frame Designations

Designation	Description
i	inertial
sr	stellar
f	planet-centered, planet-fixed
s	planet surface fixed
b	spacecraft body fixed
c	IMU case
v	velocimeter
a	altimeter
sc	star camera
t	terrain relative navigation
h	hazard relative navigation

III. Navigation Algorithm

When the system dynamics and sensors can be described by linear models (possibly time-varying) and the measurement and process noises can be described by zero-mean, white noise processes, then the Kalman filter can be shown to be the optimal *linear* filter in terms of minimizing the mean-squared state estimation error.⁵ If additionally, the measurement and process noises can be described by zero-mean, Gaussian white noise processes, the Kalman filter is the optimal filter. However, in our case, the underlying spacecraft entry dynamics and the sensors are inherently time-varying and nonlinear. The continuous-discrete extended Kalman filter (EKF), used as the primary navigation algorithm, is a direct extension of the optimal linear Kalman filter in the nonlinear setting (see, for example, Gelb⁴ for more details). The EKF has long been the workhorse of space-based navigation systems, dating back to the early days of the Apollo missions. The EKF is a recursive data processing algorithm that is capable of asynchronous fusion of measurements from various sensors, with potentially time-varying noise characteristics. Moreover, the EKF utilizes prior knowledge regarding the state of the system and the error statistics associated with the state estimate. Additionally, the EKF is model-dependent, thus making accurate modeling of the sensors and the environment a key factor in the successful implementation. In order to accommodate the

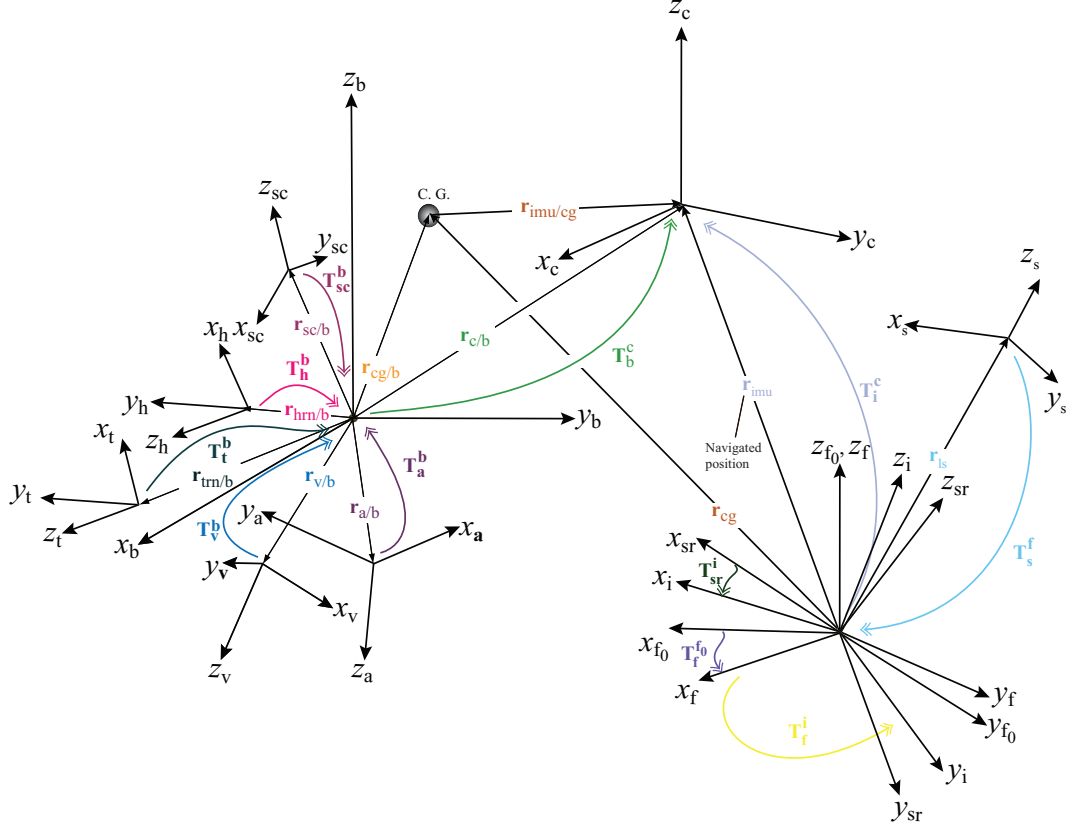


Figure 3. Reference Frame Topology.

nonlinearities of the system the EKF utilizes approximations based on Taylor series expansions of the nonlinear system dynamics and sensor models along the current estimated trajectory. With proper tuning and analysis, the EKF can provide a reasonably accurate estimate at each point along the trajectory. Tuning of the EKF can be assisted through use of *Monte Carlo* analysis that provides a means for computing the expected state estimation error covariance which can then be compared to the EKF state estimation error covariance. A well-tuned EKF will possess a state estimation error covariance that closely matches the sampled *Monte Carlo* covariance.

A. EKF Architecture

The nonlinear system model is assumed to be of the form

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t)) + \mathbf{M}(t)\mathbf{w}(t), \quad (7)$$

where $\mathbf{x}(t) \in \mathbb{R}^n$ is the state of the system, $\mathbf{f}(\mathbf{x}(t)) \in \mathbb{R}^n$ is the sufficiently differentiable nonlinear system model, and $\mathbf{w}(t) \in \mathbb{R}^p$ is a zero-mean, white-noise process with $E\{\mathbf{w}(t)\} = \mathbf{0}$ and $E\{\mathbf{w}(t)\mathbf{w}^T(\tau)\} = \mathbf{Q}_s(t)\delta(t - \tau)$ where $\mathbf{M}(t) \in \mathbb{R}^{n \times p}$ is the process noise mapping matrix, and $\mathbf{Q}_s(t) = \mathbf{Q}_s^T(t) \geq \mathbf{0} \in \mathbb{R}^{p \times p}$, $\forall t$, is the process noise spectral density. The nonlinear measurement model is assumed to be of the form

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k) + \boldsymbol{\eta}_k \quad (8)$$

where the subscript “ k ” denotes a discrete time measurement at time $t = t_k$, $\mathbf{h}_k(\mathbf{x}_k) \in \mathbb{R}^m$ is the sufficiently differentiable nonlinear measurement model evaluated at the state $\mathbf{x}_k = \mathbf{x}(t_k)$, and the measurement noise $\boldsymbol{\eta}_k \in \mathbb{R}^m$ is a zero-mean, white-noise sequence with $E\{\boldsymbol{\eta}_k\} = \mathbf{0}$ and $E\{\boldsymbol{\eta}_k\boldsymbol{\eta}_j^T\} = \mathbf{R}_k\delta_{kj}$ where $\mathbf{R}_k = \mathbf{R}_k^T > \mathbf{0} \in \mathbb{R}^{m \times m}$, $\forall t_k$. Additionally, it is assumed that the process noise and the measurement noise are not correlated in time, hence $E\{\mathbf{w}(t)\boldsymbol{\eta}_k^T\} = \mathbf{0}$, $\forall t, t_k$.

The EKF is a continuous-discrete algorithm, as illustrated in Fig. 4. The two stages are referred to as *propagation* and *update*. The propagation stage continues until one or more measurements becomes available. Once the update is completed, the propagation begins again. The state estimate is denoted by $\hat{\mathbf{x}}(t)$, the state estimation error is defined as $\mathbf{e}(t) := \mathbf{x}(t) - \hat{\mathbf{x}}(t)$, and the state estimation error covariance is $\mathbf{P}(t) := \mathbb{E}\{\mathbf{e}(t)\mathbf{e}^T(t)\}$. The state estimate and state estimation error covariance just prior to the update are denoted by $\hat{\mathbf{x}}_k^-$ and \mathbf{P}_k^- , respectively, and just after the update by $\hat{\mathbf{x}}_k^+$ and \mathbf{P}_k^+ , respectively. Note that the EKF is an *unbiased* filter, therefore, we assume that $\mathbb{E}\{\mathbf{e}(t)\} = \mathbf{0}$. For example, $\mathbf{P}_k^- = \mathbb{E}\{\mathbf{e}_k^- \mathbf{e}_k^{-T}\}$ and $\mathbf{P}_k^+ = \mathbb{E}\{\mathbf{e}_k^+ \mathbf{e}_k^{+T}\}$. The objective of the EKF is to minimize the mean-square estimation error $\mathcal{J} = \text{Tr} \mathbf{P}_k^+$, $\forall t_k$.

B. State Estimate and State Estimation Error Covariance Propagation

Consider the propagation of state estimate and the state estimation error covariance between measurements during the time interval $t \in \{t_{k-1}, t_k\}$. The EKF propagation equations are

$$\dot{\hat{\mathbf{x}}}(t) = \mathbf{f}(\hat{\mathbf{x}}(t)) \quad (9a)$$

$$\dot{\Phi}(t, t_{k-1}) = \mathbf{F}(\hat{\mathbf{x}}(t))\Phi(t, t_{k-1}) \quad (9b)$$

$$\dot{\mathbf{Q}}(t) = \mathbf{F}(\hat{\mathbf{x}}(t))\mathbf{Q}(t) + \mathbf{Q}(t)\mathbf{F}^T(\hat{\mathbf{x}}(t)) + \mathbf{M}(t)\mathbf{Q}_s(t)\mathbf{M}^T(t) \quad (9c)$$

where $t_{k-1} \leq t \leq t_k$, $\Phi(t, t_{k-1}) \in \mathbb{R}^{n \times n}$ is the state transition matrix mapping the state from t_{k-1} to t and

$$\mathbf{F}(\hat{\mathbf{x}}(t)) := \left. \frac{\partial \mathbf{f}(\mathbf{x}(t))}{\partial \mathbf{x}(t)} \right|_{\mathbf{x}(t)=\hat{\mathbf{x}}(t)} \in \mathbb{R}^{n \times n}, \quad (10)$$

with the initial conditions $\hat{\mathbf{x}}(t_{k-1}) = \hat{\mathbf{x}}_{k-1}^+$, $\Phi(t_{k-1}, t_{k-1}) = \mathbf{I}$, and $\mathbf{Q}(t_{k-1}) = \mathbf{0}$. At t_k , we have $\hat{\mathbf{x}}_k^- = \hat{\mathbf{x}}(t_k)$, $\Phi_k = \Phi(t_k, t_{k-1})$, $\mathbf{Q}_k = \mathbf{Q}(t_k)$, and $\mathbf{P}_k^- = \mathbf{P}(t_k)$. After integration, the state estimation error covariance is mapped forward via

$$\mathbf{P}_k^- = \Phi_k \mathbf{P}_{k-1}^+ \Phi_k^T + \mathbf{Q}_k. \quad (11)$$

C. State Estimate and State Estimation Error Covariance Update

The state estimate and the state estimation error covariance update are calculated via

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{y}_k - \mathbf{h}_k(\hat{\mathbf{x}}_k^-)) \quad (12a)$$

$$\mathbf{P}_k^+ = [\mathbf{I} - \mathbf{K}_k \mathbf{H}_k(\hat{\mathbf{x}}_k^-)] \mathbf{P}_k^- [\mathbf{I} - \mathbf{K}_k \mathbf{H}_k(\hat{\mathbf{x}}_k^-)]^T + \mathbf{K}_k \mathbf{R}_k \mathbf{K}_k^T \quad (12b)$$

where the Kalman gain is computed via

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T(\hat{\mathbf{x}}_k^-) \mathbf{W}_k^{-1}, \quad (13)$$

where \mathbf{W}_k is the measurement residual covariance matrix given by

$$\mathbf{W}_k = \mathbf{H}_k(\hat{\mathbf{x}}_k^-) \mathbf{P}_k^- \mathbf{H}_k^T(\hat{\mathbf{x}}_k^-) + \mathbf{R}_k. \quad (14)$$

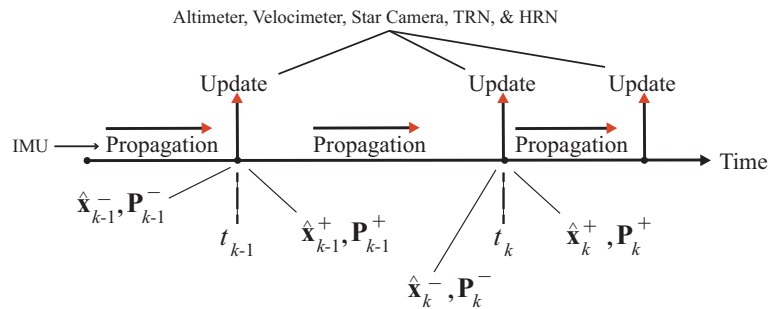


Figure 4. EKF timeline.

and where $\mathbf{H}_k(\hat{\mathbf{x}}_k^-)$ is the measurement sensitivity matrix defined to be

$$\mathbf{H}_k(\hat{\mathbf{x}}_k^-) = \left. \frac{\partial \mathbf{h}_k(\mathbf{x}_k)}{\partial \mathbf{x}_k} \right|_{\mathbf{x}_k = \hat{\mathbf{x}}_k^-} \in \mathbb{R}^{m \times n}. \quad (15)$$

IV. Dynamics Modeling

Extended Kalman filter navigation algorithms are model-based. This implies that the models of the translational and rotational dynamics of the lander are key to accurately predicting the motion of the spacecraft. Furthermore, without an accurate measure of the non-gravitational acceleration and the spacecraft angular velocity as provided by an accelerometer/gyro package, the translational and rotational dynamics models would not be capable of supporting precision landing. The navigation algorithm must also possess accelerometer and gyro models. Also of interest in precision landing is the location of the landing site. In performing precision descent-to-landing navigation, it is necessary to be able to accurately predict the location of the landing site, accounting for the presence of map tie errors. In addition to the dynamical models describing translation and rotation of the lander, it is necessary to formulate the dynamics of the estimation errors. The estimation errors represent the differences between the true spacecraft state (position, velocity, and attitude) and the navigation provided estimated state. Other error dynamics associated with systematic errors are also included in the development.

The total acceleration of the vehicle is comprised of accelerations due to conservative forces (e.g., gravity), and accelerations due to non-conservative forces. The non-conservative forces acting upon the vehicle may include, but are not necessarily limited to, forces due to engine thrust and forces due to atmospheric interaction with the vehicle. An IMU consisting of a set of accelerometers and gyros senses the linear non-gravitational acceleration and the rotational rate of the vehicle, respectively. We navigate the IMU navigation base (rather than the center-of-gravity). The translational and rotational dynamics are

$$\dot{\mathbf{r}}_{imu}^i(t) = \mathbf{v}_{imu}^i(t) \quad (16a)$$

$$\dot{\mathbf{v}}_{imu}^i(t) = \mathbf{T}_f^i(t) \mathbf{a}_g(\mathbf{r}_{cg}^f(t)) + \mathbf{T}_b^i(\bar{\mathbf{q}}_i^b(t)) \mathbf{T}_c^b \mathbf{a}_{ng}^c(t) \quad (16b)$$

$$\dot{\bar{\mathbf{q}}}_i^b(t) = \frac{1}{2} \bar{\boldsymbol{\omega}}_{b/i}^b(t) \otimes \bar{\mathbf{q}}_i^b(t) \quad (16c)$$

where $\mathbf{r}_{imu}^i(t)$ and $\mathbf{v}_{imu}^i(t)$ are the position and velocity, respectively, of the IMU represented in the inertial reference frame, $\bar{\mathbf{q}}_i^b(t)$ represents the orientation of the spacecraft body fixed reference frame with respect to the inertial reference frame, and

$$\mathbf{r}_{cg}^f(t) = \mathbf{T}_i^f(t) \mathbf{r}_{imu}^i(t) + \mathbf{T}_i^f(t) \mathbf{T}_b^i(\bar{\mathbf{q}}_i^b(t)) \mathbf{r}_{cg/imu}^b(t) \quad (17a)$$

$$\bar{\boldsymbol{\omega}}_{b/i}^b(t) = \begin{pmatrix} \mathbf{T}_c^b \boldsymbol{\omega}_{b/i}^c(t) \\ 0 \end{pmatrix}, \quad (17b)$$

where the non-gravitational acceleration, $\mathbf{a}_{ng}^c(t)$, and body rate, $\boldsymbol{\omega}_{b/i}^c(t)$, are sensed by the IMU in the IMU case reference frame, and $\mathbf{a}_g(\mathbf{r}_{cg}^f(t))$ is the gravitational acceleration represented in the planet-centered, planet-fixed reference frame, $\mathbf{r}_{cg/imu}^b(t)$ is the position of the center-of-gravity relative to the IMU represented in the body fixed reference frame. The location of the IMU relative to the center-of-gravity is given by a nominal position, denoted by $\bar{\mathbf{r}}_{cg/imu}^b(t)$, in the body fixed reference frame and computed externally to the navigation system (and likely time-varying as fuel is expended during descent) and a deviation from the nominal position, denoted by $\Delta \mathbf{r}_{cg/imu}^b(t)$ such that

$$\mathbf{r}_{cg/imu}^b(t) = \bar{\mathbf{r}}_{cg/imu}^b(t) + \Delta \mathbf{r}_{cg/imu}^b(t). \quad (18)$$

Also, we have $\mathbf{T}_b^i(\bar{\mathbf{q}}_i^b(t)) = [\mathbf{T}_i^b(\bar{\mathbf{q}}_i^b(t))]^T$ where

$$\mathbf{T}_i^b(\bar{\mathbf{q}}_i^b(t)) = \mathbf{I}_{3 \times 3} - 2\bar{q}_i^b(t) [\bar{\mathbf{q}}_i^b(t) \times] + 2 [\bar{\mathbf{q}}_i^b(t) \times]^2. \quad (19)$$

The transformation matrices $\mathbf{T}_f^i(t)$ and \mathbf{T}_b^c are assumed known.

A. Gravity Modeling

There are various possibilities for modeling the gravitational acceleration, $\mathbf{a}_g(\mathbf{r}_{cg}^f(t))$ in Eq. (16). Typically, we model the primary gravitational body using point-mass approximations and zonal harmonic approximations.^{6-8,10} The simplest model assumes a spherical planet, and

$$\mathbf{a}_g(\mathbf{r}_{cg}^f(t)) = -\frac{\mu}{r_{cg}^3(t)} \mathbf{r}_{cg}^f(t), \quad (20)$$

where $r_{cg}(t) = \|\mathbf{r}_{cg}^f(t)\| \in \mathbb{R}$ and μ is the gravitational parameter. The associated gravity gradient matrix is

$$\mathbf{G}(\mathbf{r}_{cg}^f(t)) = \frac{\mu}{r_{cg}^5(t)} \left(3\mathbf{r}_{cg}^f(t)(\mathbf{r}_{cg}^f(t))^T - r_{cg}^2(t)\mathbf{I}_{3 \times 3} \right) \in \mathbb{R}^{3 \times 3}. \quad (21)$$

The gravity gradient matrix is used in propagation of the estimation error equations.

B. Inertial Measurement Unit

1. Accelerometers

The IMU measurement of the non-gravitational acceleration is corrupted by errors due to nonorthogonality and misalignment of the axes, $\mathbf{\Gamma}_a \in \mathbb{R}^{3 \times 3}$, errors due to scale-factor uncertainties, $\mathbf{S}_a \in \mathbb{R}^{3 \times 3}$, random biases, $\mathbf{b}_a \in \mathbb{R}^3$, and noise, $\boldsymbol{\eta}_a(t) \in \mathbb{R}^3$. Incorporating these error sources, the measured non-gravitational acceleration, $\mathbf{a}_{ng,m}^c(t) \in \mathbb{R}^3$, can be written in terms of the true non-gravitational acceleration, $\mathbf{a}_{ng}^c(t) \in \mathbb{R}^3$, as^{11,12}

$$\mathbf{a}_{ng,m}^c(t) = (\mathbf{I}_{3 \times 3} + \mathbf{\Gamma}_a) (\mathbf{I}_{3 \times 3} + \mathbf{S}_a) (\mathbf{a}_{ng}^c(t) + \mathbf{b}_a + \boldsymbol{\eta}_a(t)), \quad (22)$$

where

$$\mathbf{\Gamma}_a := \begin{bmatrix} 0 & \gamma_{a_{xz}} & -\gamma_{a_{xy}} \\ -\gamma_{a_{yz}} & 0 & \gamma_{a_{yx}} \\ \gamma_{a_{zy}} & -\gamma_{a_{zx}} & 0 \end{bmatrix}, \quad \mathbf{S}_a := \begin{bmatrix} s_{a_x} & 0 & 0 \\ 0 & s_{a_y} & 0 \\ 0 & 0 & s_{a_z} \end{bmatrix},$$

$$\mathbf{a}_{ng,m}^c(t) := \begin{pmatrix} a_{m_x}^c(t) \\ a_{m_y}^c(t) \\ a_{m_z}^c(t) \end{pmatrix}, \quad \mathbf{a}_{ng}^c(t) := \begin{pmatrix} a_x^c(t) \\ a_y^c(t) \\ a_z^c(t) \end{pmatrix}, \quad \mathbf{b}_a := \begin{pmatrix} b_{a_x} \\ b_{a_y} \\ b_{a_z} \end{pmatrix}, \quad \boldsymbol{\eta}_a(t) := \begin{pmatrix} \eta_{a_x}(t) \\ \eta_{a_y}(t) \\ \eta_{a_z}(t) \end{pmatrix},$$

and where $\boldsymbol{\eta}_a(t)$ is a zero-mean, white noise process with covariance

$$\mathbb{E} \{ \boldsymbol{\eta}_a(t) \boldsymbol{\eta}_a^T(\tau) \} = \mathbf{Q}_a \delta(t - \tau) \in \mathbb{R}^{3 \times 3},$$

and $\mathbf{Q}_a = \mathbf{Q}_a^T > 0$. To first-order, the non-gravitational acceleration in Eq. (26) may be written in terms of the measured non-gravitational acceleration and the model parameters as

$$\mathbf{a}_{ng}^c(t) = \mathbf{a}_{ng,m}^c(t) - \mathbf{M}_1(\mathbf{a}_{ng,m}^c(t))\boldsymbol{\gamma}_a - \mathbf{M}_2(\mathbf{a}_{ng,m}^c(t))\mathbf{s}_a - \mathbf{b}_a - \boldsymbol{\eta}_a(t). \quad (23)$$

where

$$\mathbf{M}_1(\mathbf{a}_{ng,m}^c(t)) := \begin{bmatrix} -a_{m_z}^c(t) & a_{m_y}^c(t) & 0 & 0 & 0 & 0 \\ 0 & 0 & a_{m_z}^c(t) & -a_{m_x}^c(t) & 0 & 0 \\ 0 & 0 & 0 & 0 & -a_{m_y}^c(t) & a_{m_x}^c(t) \end{bmatrix},$$

$$\mathbf{M}_2(\mathbf{a}_{ng,m}^c(t)) := \begin{bmatrix} a_{m_x}^c(t) & 0 & 0 \\ 0 & a_{m_y}^c(t) & 0 \\ 0 & 0 & a_{m_z}^c(t) \end{bmatrix}, \quad \boldsymbol{\gamma}_a := \begin{pmatrix} \gamma_{a_{xy}} \\ \gamma_{a_{xz}} \\ \gamma_{a_{yx}} \\ \gamma_{a_{yz}} \\ \gamma_{a_{zx}} \\ \gamma_{a_{zy}} \end{pmatrix}, \quad \text{and } \mathbf{s}_a := \begin{pmatrix} s_{a_x} \\ s_{a_y} \\ s_{a_z} \end{pmatrix}.$$

We model \mathbf{b}_a , γ_a , and \mathbf{s}_a as unknown constants with

$$\dot{\mathbf{b}}_a = 0, \quad \dot{\gamma}_a = 0, \dot{\mathbf{s}}_a = 0, \quad (24)$$

and $E\{\mathbf{b}_a\} = 0$, $E\{\gamma_a\} = 0$, $E\{\mathbf{s}_a\} = 0$, $E\{\mathbf{b}_a \mathbf{b}_a^T\} = \mathbf{P}_{\mathbf{b}_a} = \mathbf{P}_{\mathbf{b}_a}^T \in \mathbb{R}^{3 \times 3}$, $E\{\mathbf{s}_a \mathbf{s}_a^T\} = \mathbf{P}_{\mathbf{s}_a} = \mathbf{P}_{\mathbf{s}_a}^T \in \mathbb{R}^{3 \times 3}$, and $E\{\gamma_a \gamma_a^T\} = \mathbf{P}_{\gamma_a} = \mathbf{P}_{\gamma_a}^T \in \mathbb{R}^{6 \times 6}$, and where $\mathbf{P}_{\mathbf{b}_a} > 0$, $\mathbf{P}_{\mathbf{s}_a} > 0$, and $\mathbf{P}_{\gamma_a} > 0$ are given. In the EKF implementation, we often model the unknown constants as random constants driven by zero-mean, white noise processes with given process noise strength or as exponentially correlated random variables with given time constants and process noise strengths (see Gelb⁴). For example, the dynamics of the unknown constants in Eq. (24) can be modeled as

$$\dot{\mathbf{b}}_a(t) = \boldsymbol{\eta}_{b_a}(t), \quad \dot{\gamma}_a(t) = \boldsymbol{\eta}_{\gamma_a}(t), \dot{\mathbf{s}}_a(t) = \boldsymbol{\eta}_{s_a}(t), \quad (25)$$

where $E\{\boldsymbol{\eta}_{b_a}(t)\} = 0$, $E\{\boldsymbol{\eta}_{\gamma_a}(t)\} = 0$, $E\{\boldsymbol{\eta}_{s_a}(t)\} = 0$, for $\forall t$, $E\{\boldsymbol{\eta}_{b_a}(t) \boldsymbol{\eta}_{b_a}^T(\tau)\} = \mathbf{Q}_{b_a} \delta(t - \tau)$, $E\{\boldsymbol{\eta}_{s_a}(t) \boldsymbol{\eta}_{s_a}^T(\tau)\} = \mathbf{Q}_{s_a} \delta(t - \tau)$, and $E\{\boldsymbol{\eta}_{\gamma_a}(t) \boldsymbol{\eta}_{\gamma_a}^T(\tau)\} = \mathbf{Q}_{\gamma_a} \delta(t - \tau)$, for $\forall t, \tau$ and where $\mathbf{Q}_{b_a} = \mathbf{Q}_{b_a}^T \in \mathbb{R}^{3 \times 3}$, $\mathbf{Q}_{s_a} = \mathbf{Q}_{s_a}^T \in \mathbb{R}^{3 \times 3}$, $\mathbf{Q}_{\gamma_a} = \mathbf{Q}_{\gamma_a}^T \in \mathbb{R}^{6 \times 6}$, where $\mathbf{Q}_{b_a} \geq 0$, $\mathbf{Q}_{s_a} \geq 0$, and $\mathbf{Q}_{\gamma_a} \geq 0$ are given.

2. Gyros

The IMU measurement of the angular velocity of the spacecraft is corrupted by errors due to nonorthogonality and misalignment of the axes, $\boldsymbol{\Gamma}_g \in \mathbb{R}^{3 \times 3}$, errors due to scale-factor uncertainties, $\mathbf{S}_g \in \mathbb{R}^{3 \times 3}$, random biases, $\mathbf{b}_g \in \mathbb{R}^3$, and noise, $\boldsymbol{\eta}_g(t) \in \mathbb{R}^3$. Incorporating these error sources, the measured angular velocity, $\boldsymbol{\omega}_{b/i,m}^c(t) \in \mathbb{R}^3$, can be written in terms of the true angular velocity, $\boldsymbol{\omega}_{b/i}^c(t) \in \mathbb{R}^3$, as

$$\boldsymbol{\omega}_{b/i,m}^c(t) = (\mathbf{I}_{3 \times 3} + \boldsymbol{\Gamma}_g) (\mathbf{I}_{3 \times 3} + \mathbf{S}_g) \left(\boldsymbol{\omega}_{b/i}^c(t) + \mathbf{b}_g + \boldsymbol{\eta}_g(t) \right), \quad (26)$$

where

$$\boldsymbol{\Gamma}_g := \begin{bmatrix} 0 & \gamma_{g_{xz}} & -\gamma_{g_{xy}} \\ -\gamma_{g_{yz}} & 0 & \gamma_{g_{yx}} \\ \gamma_{g_{zy}} & -\gamma_{g_{zx}} & 0 \end{bmatrix}, \quad \mathbf{S}_g := \begin{bmatrix} s_{g_x} & 0 & 0 \\ 0 & s_{g_y} & 0 \\ 0 & 0 & s_{g_z} \end{bmatrix},$$

$$\boldsymbol{\omega}_{b/i,m}^c(t) := \begin{pmatrix} \omega_{m_x}^c(t) \\ \omega_{m_y}^c(t) \\ \omega_{m_z}^c(t) \end{pmatrix}, \boldsymbol{\omega}_{b/i}^c(t) := \begin{pmatrix} \omega_x^c(t) \\ \omega_y^c(t) \\ \omega_z^c(t) \end{pmatrix}, \mathbf{b}_g := \begin{pmatrix} b_{g_x} \\ b_{g_y} \\ b_{g_z} \end{pmatrix}, \boldsymbol{\eta}_g(t) := \begin{pmatrix} \eta_{g_x}(t) \\ \eta_{g_y}(t) \\ \eta_{g_z}(t) \end{pmatrix},$$

and where $\boldsymbol{\eta}_g(t)$ is a zero-mean, white noise process with covariance

$$E\{\boldsymbol{\eta}_g(t) \boldsymbol{\eta}_g^T(\tau)\} = \mathbf{Q}_g \delta(t - \tau) \in \mathbb{R}^{3 \times 3},$$

and $\mathbf{Q}_g = \mathbf{Q}_g^T > 0$ is given. To first-order, the angular velocity in Eq. (26) may be written in terms of the measured angular velocity and the model parameters as

$$\boldsymbol{\omega}_{b/i}^c(t) = \boldsymbol{\omega}_{b/i,m}^c(t) - \mathbf{N}_1(\boldsymbol{\omega}_{b/i,m}^c(t)) \boldsymbol{\gamma}_g - \mathbf{N}_2(\boldsymbol{\omega}_{b/i,m}^c(t)) \mathbf{s}_g - \mathbf{b}_g - \boldsymbol{\eta}_g(t). \quad (27)$$

where

$$\mathbf{N}_1(\boldsymbol{\omega}_{b/i,m}^c(t)) := \begin{bmatrix} -\omega_{m_z}^c(t) & \omega_{m_y}^c(t) & 0 & 0 & 0 & 0 \\ 0 & 0 & \omega_{m_z}^c(t) & -\omega_{m_x}^c(t) & 0 & 0 \\ 0 & 0 & 0 & 0 & -\omega_{m_y}^c(t) & \omega_{m_x}^c(t) \end{bmatrix},$$

$$\mathbf{N}_2(\boldsymbol{\omega}_{b/i,m}^c(t)) := \begin{bmatrix} \omega_{m_x}^c(t) & 0 & 0 \\ 0 & \omega_{m_y}^c(t) & 0 \\ 0 & 0 & \omega_{m_z}^c(t) \end{bmatrix}, \boldsymbol{\gamma}_g := \begin{pmatrix} \gamma_{g_{xy}} \\ \gamma_{g_{xz}} \\ \gamma_{g_{yx}} \\ \gamma_{g_{yz}} \\ \gamma_{g_{zx}} \\ \gamma_{g_{zy}} \end{pmatrix}, \text{ and } \mathbf{s}_g := \begin{pmatrix} s_{g_x} \\ s_{g_y} \\ s_{g_z} \end{pmatrix}.$$

We model \mathbf{b}_g , γ_g , and \mathbf{s}_g as unknown constants with

$$\dot{\mathbf{b}}_g = 0, \quad \dot{\gamma}_g = 0, \quad \text{and} \quad \dot{\mathbf{s}}_g = 0, \quad (28)$$

and $E\{\mathbf{b}_g\} = 0$, $E\{\gamma_g\} = 0$, $E\{\mathbf{s}_g\} = 0$, and $E\{\mathbf{b}_g \mathbf{b}_g^T\} = \mathbf{P}_{\mathbf{b}_g} = \mathbf{P}_{\mathbf{b}_g}^T \in \mathbb{R}^{3 \times 3}$, $E\{\mathbf{s}_g \mathbf{s}_g^T\} = \mathbf{P}_{\mathbf{s}_g} = \mathbf{P}_{\mathbf{s}_g}^T \in \mathbb{R}^{3 \times 3}$, and $E\{\gamma_g \gamma_g^T\} = \mathbf{P}_{\gamma_g} = \mathbf{P}_{\gamma_g}^T \in \mathbb{R}^{6 \times 6}$ and where $\mathbf{P}_{\mathbf{b}_g} > 0$, $\mathbf{P}_{\mathbf{s}_g} > 0$, and $\mathbf{P}_{\gamma_g} > 0$ are given. As with the accelerometer unknown constants in Eq. (24), we can model the unknown gyro constants in Eq. (28) as random constants driven by zero-mean, white noise processes with given process noise strength or as exponentially correlated random variables with given time constants and process noise strengths.

C. Landing Site

The nominal landing site position in the inertial reference frame is

$$\mathbf{r}_{ls}^i(t) = \mathbf{T}_f^i(t) \mathbf{r}_{ls}^f, \quad (29)$$

where \mathbf{r}_{ls}^f is the fixed position of the landing site in the planet-centered, planet-fixed reference frame and $\mathbf{T}_f^i(t)$ is the known transformation matrix from the planet-centered, planet-fixed reference frame to the inertial reference frame. The planet is modeled as an oblate spheroid (or ellipsoid), and therefore the geodetic coordinate system is used to represent the landing site as^{6,8}

$$\mathbf{r}_{ls}^f = \begin{bmatrix} (N+h) \cos \phi \cos \lambda \\ (N+h) \cos \phi \sin \lambda \\ (N(1-e^2)+h) \sin \phi \end{bmatrix}, \quad (30)$$

where h is the altitude above the reference ellipsoid, ϕ is the geodetic latitude, λ is the longitude, e is the first eccentricity, and N is the radius of curvature in the prime vertical given by

$$N = \frac{R_e}{\sqrt{1 - e^2 \sin^2 \phi}}, \quad (31)$$

where R_e is the equatorial radius. While the surface features may be well-known relative to one another, the precise location of surface features (including the desired landing site) in the planet-centered, planet-fixed reference frame may possess map-tie errors, and therefore the location of the surface point as represented in the inertial reference frame possesses uncertainties. We model the true altitude, geodetic latitude, and longitude in Eq. (30) and Eq. (31) as

$$h = h_{ls} + h_{\Delta}, \quad \phi = \phi_{ls} + \phi_{\Delta}, \quad \text{and} \quad \lambda = \lambda_{ls} + \lambda_{\Delta}, \quad (32)$$

where the nominal altitude above the reference ellipsoid, geodetic latitude, and longitude of the landing site, denoted by h_{ls} , ϕ_{ls} , and λ_{ls} , respectively, are assumed known *a priori*, and the associated map-tie errors are denoted by h_{Δ} , ϕ_{Δ} and λ_{Δ} , respectively. We model the map-tie errors are unknown constants with

$$\dot{h}_{\Delta} = 0, \quad \dot{\phi}_{\Delta} = 0, \quad \dot{\lambda}_{\Delta} = 0, \quad (33)$$

where $E\{h_{\Delta}\} = 0$, $E\{\phi_{\Delta}\} = 0$, $E\{\lambda_{\Delta}\} = 0$, and $E\{h_{\Delta}^2\} = \mathbf{P}_{h_{\Delta}} > 0 \in \mathbb{R}$, $E\{\phi_{\Delta}^2\} = \mathbf{P}_{\phi_{\Delta}} > 0 \in \mathbb{R}$, and $E\{\lambda_{\Delta}^2\} = \mathbf{P}_{\lambda_{\Delta}} > 0 \in \mathbb{R}$ are given.

D. EKF State Dynamics Model

The state dynamics model in the navigation algorithm follows from Eq. (16)–Eq. (32). The EKF state dynamics are given by

$$\dot{\hat{\mathbf{r}}}_{imu}^i(t) = \hat{\mathbf{v}}_{imu}^i(t) \quad (34a)$$

$$\dot{\hat{\mathbf{v}}}_{imu}^i(t) = \mathbf{T}_f^i(t) \mathbf{a}_g(\hat{\mathbf{r}}_{cg}^f(t)) + \mathbf{T}_b^i(\hat{\mathbf{q}}_i^b(t)) \mathbf{T}_c^b \hat{\mathbf{a}}_{ng}^c(t) \quad (34b)$$

$$\dot{\hat{\mathbf{q}}}_i^b(t) = \frac{1}{2} \hat{\boldsymbol{\omega}}_{b/i}^b(t) \otimes \hat{\mathbf{q}}_i^b(t) \quad (34c)$$

$$\dot{\hat{\mathbf{b}}}_a = 0, \dot{\hat{\gamma}}_a = 0, \dot{\hat{\mathbf{s}}}_a = 0, \dot{\hat{\mathbf{b}}}_g = 0, \dot{\hat{\gamma}}_g = 0, \dot{\hat{\mathbf{s}}}_g = 0, \dot{\hat{h}}_{\Delta} = 0, \dot{\hat{\phi}}_{\Delta} = 0, \dot{\hat{\lambda}}_{\Delta} = 0, \quad (34d)$$

where

$$\mathbf{r}_{cg}^f(t) = \mathbf{T}_i^f(t)\mathbf{r}_{imu}^i(t) + \mathbf{T}_i^f(t)\mathbf{T}_b^i(\hat{\mathbf{q}}_i^b(t))\mathbf{r}_{cg/imu}^b(t) \quad (35a)$$

$$\mathbf{r}_{cg/imu}^b(t) = \bar{\mathbf{r}}_{cg/imu}^b(t) + \Delta\mathbf{r}_{cg/imu}^b(t) \quad (35b)$$

$$\mathbf{T}_i^b(\hat{\mathbf{q}}_i^b(t)) = \mathbf{I}_{3 \times 3} - 2\hat{q}_i^b(t) [\hat{\mathbf{q}}_i^b(t) \times] + 2 [\hat{\mathbf{q}}_i^b(t) \times]^2 \quad (35c)$$

$$\hat{\omega}_{b/i}^b(t) = \begin{pmatrix} \mathbf{T}_c^b \hat{\omega}_{b/i}^c(t) \\ 0 \end{pmatrix} \quad (35d)$$

and

$$\hat{\mathbf{a}}_{ng}^c(t) = \mathbf{a}_{ng,m}^c(t) - \mathbf{M}_1(\mathbf{a}_{ng,m}^c)\hat{\gamma}_a - \mathbf{M}_2(\mathbf{a}_{ng,m}^c)\hat{\mathbf{s}}_a - \hat{\mathbf{b}}_a \quad (36a)$$

$$\hat{\omega}_{b/i}^c(t) = \omega_{b/i,m}^c(t) - \mathbf{N}_1(\omega_{b/i,m}^c)\hat{\gamma}_g - \mathbf{N}_2(\omega_{b/i,m}^c)\hat{\mathbf{s}}_g - \hat{\mathbf{b}}_g \quad (36b)$$

V. Estimation Error Dynamics

A. Attitude Estimation Errors

Define the attitude estimation error as

$$\delta\bar{\mathbf{q}}_i^b(t) := \bar{\mathbf{q}}_i^b(t) \otimes [\hat{\mathbf{q}}_i^b(t)]^{-1}. \quad (37)$$

Taking the time derivative of the error quaternion in Eq. (37), substituting $\dot{\bar{\mathbf{q}}}_i^b(t)$ and $\dot{\hat{\mathbf{q}}}_i^b(t)$ from Eq. (16c) and Eq. (34c) respectively, using the definition of quaternion multiplication, and rearranging yields

$$\delta\dot{\bar{\mathbf{q}}}_i^b(t) = \frac{1}{2} \begin{bmatrix} \delta q_i^b(t)\delta\omega_{b/i}^b(t) - 2\hat{\omega}_{b/i}^b(t) \times \delta\mathbf{q}_i^b(t) - \delta\omega_{b/i}^b(t) \times \delta\mathbf{q}_i^b(t) \\ -\delta\omega_{b/i}^b(t) \odot \delta\mathbf{q}_i^b(t) \end{bmatrix}, \quad (38)$$

where $\delta\omega_{b/i}^b(t) := \omega_{b/i}^b(t) - \hat{\omega}_{b/i}^b(t) = \mathbf{T}_c^b(\omega_{b/i}^c(t) - \hat{\omega}_{b/i}^c(t))$. Assuming small attitude errors and neglecting second-order terms, Eq. (38) reduces to

$$\begin{aligned} \delta\dot{\bar{\mathbf{q}}}_i^b(t) &= -\hat{\omega}_{b/i}^b(t) \times \delta\mathbf{q}_i^b(t) + \frac{1}{2}\delta\omega_{b/i}^b(t) \\ \delta\dot{q}_i^b(t) &= 0. \end{aligned} \quad (39)$$

Define $\delta\mathbf{b}_g := \mathbf{b}_g - \hat{\mathbf{b}}_g$, $\delta\mathbf{s}_g := \mathbf{s}_g - \hat{\mathbf{s}}_g$, and $\delta\gamma_g := \gamma_g - \hat{\gamma}_g$. Computing $\delta\omega_{b/i}^b(t)$ using Eq. (27) and Eq. (36b) yields

$$\delta\omega_{b/i}^b(t) = -\mathbf{T}_c^b \left[\mathbf{N}_1(\omega_{b/i,m}^c(t))\delta\gamma_g + \mathbf{N}_2(\omega_{b/i,m}^c(t))\delta\mathbf{s}_g + \delta\mathbf{b}_g + \boldsymbol{\eta}_g(t) \right]. \quad (40)$$

Substituting Eq. (40) into Eq. (39) yields

$$\begin{aligned} \delta\dot{\bar{\mathbf{q}}}_i^b(t) &= - \left[\mathbf{T}_c^b(\omega_{b/i,m}^c(t)) \times \right] \delta\mathbf{q}_i^b(t) - \frac{1}{2}\mathbf{T}_c^b \left(\mathbf{N}_1(\omega_{b/i,m}^c(t))\delta\gamma_g \right. \\ &\quad \left. + \mathbf{N}_2(\omega_{b/i,m}^c(t))\delta\mathbf{s}_g + \delta\mathbf{b}_g + \boldsymbol{\eta}_g(t) \right). \end{aligned} \quad (41)$$

Assuming small attitude error angles, denoted by $\delta\boldsymbol{\alpha}$, we have the approximation

$$\delta\mathbf{q}_i^b(t) \approx \frac{1}{2}\delta\boldsymbol{\alpha}(t). \quad (42)$$

Taking the time-derivative of Eq. (42) and substituting Eq. (41) yields

$$\delta\dot{\boldsymbol{\alpha}}(t) = - \left[\mathbf{T}_c^b(\omega_{b/i,m}^c(t)) \times \right] \delta\boldsymbol{\alpha}(t) - \mathbf{T}_c^b \left(\mathbf{N}_1(\omega_{b/i,m}^c(t))\delta\gamma_g + \mathbf{N}_2(\omega_{b/i,m}^c(t))\delta\mathbf{s}_g + \delta\mathbf{b}_g + \boldsymbol{\eta}_g(t) \right). \quad (43)$$

With the attitude estimation error as in Eq. (37) and the approximation in Eq. (42), we have the relationship

$$\mathbf{T}_b^i(\bar{\mathbf{q}}_i^b(t)) \approx \mathbf{T}_b^i(\hat{\mathbf{q}}_i^b(t)) \left[\mathbf{I} + [\delta\boldsymbol{\alpha} \times] \right]. \quad (44)$$

B. Position and Velocity Estimation Errors

Define the position estimation error and velocity estimation as $\delta \mathbf{r}_{imu}^i(t) := \mathbf{r}_{imu}^i(t) - \hat{\mathbf{r}}_{imu}^i(t)$ and $\delta \mathbf{v}_{imu}^i(t) := \mathbf{v}_{imu}^i(t) - \hat{\mathbf{v}}_{imu}^i(t)$, respectively. Taking the time-derivative, using the relationships in Eq. (16) and Eq. (34), and collecting terms yields

$$\delta \dot{\mathbf{r}}_{imu}^i(t) = \delta \mathbf{v}_{imu}^i(t) \quad (45a)$$

$$\delta \dot{\mathbf{v}}_{imu}^i(t) = \mathbf{T}_f^i(t) (\mathbf{a}_g(\mathbf{r}_{cg}^f(t)) - \mathbf{a}_g(\hat{\mathbf{r}}_{cg}^f(t))) + \mathbf{T}_b^i(\hat{\mathbf{q}}_i^b(t)) \mathbf{T}_c^b \mathbf{a}_{ng}^c(t) - \mathbf{T}_b^i(\hat{\mathbf{q}}_i^b(t)) \mathbf{T}_c^b \hat{\mathbf{a}}_{ng}^c(t). \quad (45b)$$

Expanding the acceleration due to gravity in a Taylor series and keeping the first-order terms yields the approximation

$$\mathbf{a}_g(\mathbf{r}_{cg}^f(t)) - \mathbf{a}_g(\hat{\mathbf{r}}_{cg}^f(t)) \approx \mathbf{G}(\hat{\mathbf{r}}_{cg}^f(t)) \delta \mathbf{r}_{cg}^f(t), \quad (46)$$

where the gravity gradient matrix $\mathbf{G}(\mathbf{r}_{cg}^f(t))$ is given in Eq. (21) and where we define $\delta \mathbf{r}_{cg}^f(t) := \mathbf{r}_{cg}^f(t) - \hat{\mathbf{r}}_{cg}^f(t)$. Note that we implicitly assume that the true gravity has the same functional form as the model employed in the EKF—in this case, a spherical gravity model. In reality, the true gravity is significantly more complex; however, this approach has proven to be effective in our EKF filter implementation. More complex gravity models can certainly be employed with the associated additional complexity in the state and state estimation error covariance propagation. Using Eq. (17), Eq. (18), Eq. (35a), and the definition $\delta \mathbf{r}_{cg/imu}^b(t) := \Delta \mathbf{r}_{cg/imu}^b(t) - \Delta \hat{\mathbf{r}}_{cg/imu}^b(t)$, it can be shown that

$$\delta \mathbf{r}_{cg}^f(t) = \mathbf{T}_i^f(t) \delta \mathbf{r}_{imu}^i(t) - \mathbf{T}_i^f(t) \mathbf{T}_b^i(\hat{\mathbf{q}}_i^b(t)) \left[\hat{\mathbf{r}}_{cg/imu}^b \times \right] \delta \boldsymbol{\alpha}(t) + \mathbf{T}_b^i(\hat{\mathbf{q}}_i^b(t)) \delta \mathbf{r}_{cg/imu}^b(t). \quad (47)$$

Then from Eq. (27), Eq. (36), Eq. (44) it follows that to first-order we have

$$\begin{aligned} \mathbf{T}_b^i(\hat{\mathbf{q}}_i^b(t)) \mathbf{T}_c^b \mathbf{a}_{ng}^c(t) - \mathbf{T}_b^i(\hat{\mathbf{q}}_i^b(t)) \mathbf{T}_c^b \hat{\mathbf{a}}_{ng}^c(t) &= -\mathbf{T}_b^i(\hat{\mathbf{q}}_i^b(t)) \left[\mathbf{T}_c^b \mathbf{a}_{ng,m}^c(t) \times \right] \delta \boldsymbol{\alpha}(t) \\ &\quad - \mathbf{T}_b^i(\hat{\mathbf{q}}_i^b(t)) \mathbf{T}_c^b (\delta \mathbf{b}_a(t) - \mathbf{M}_1(\mathbf{a}_{ng,m}^c(t)) \delta \boldsymbol{\gamma}_a(t) - \mathbf{M}_2(\mathbf{a}_{ng,m}^c(t)) \delta \mathbf{s}_a(t) - \boldsymbol{\eta}_a(t)) \\ &\quad - \mathbf{G}(\hat{\mathbf{r}}_{cg}^f(t)) \mathbf{T}_b^i(\hat{\mathbf{q}}_i^b(t)) \delta \mathbf{r}_{cg/imu}^b(t), \end{aligned} \quad (48)$$

where $\delta \mathbf{b}_a = \mathbf{b}_a - \hat{\mathbf{b}}_a$, $\delta \mathbf{s}_a = \mathbf{s}_a - \hat{\mathbf{s}}_a$, and $\delta \boldsymbol{\gamma}_a = \boldsymbol{\gamma}_a - \hat{\boldsymbol{\gamma}}_a$. Substituting Eq. (47) and Eq. (48) into Eq. (45b) yields

$$\begin{aligned} \delta \dot{\mathbf{v}}_{imu}^i(t) &= \mathbf{G}(\hat{\mathbf{r}}_{cg}^f(t)) \delta \mathbf{r}_{imu}^i(t) - \mathbf{G}(\hat{\mathbf{r}}_{cg}^f(t)) \mathbf{T}_b^i(\hat{\mathbf{q}}_i^b(t)) \delta \mathbf{r}_{cg/imu}^b(t) \\ &\quad - \left[\mathbf{T}_b^i(\hat{\mathbf{q}}_i^b(t)) \left[\mathbf{T}_c^b \mathbf{a}_{ng,m}^c(t) \times \right] + \mathbf{G}(\hat{\mathbf{r}}_{cg}^f(t)) \mathbf{T}_b^i(\hat{\mathbf{q}}_i^b(t)) \left[\hat{\mathbf{r}}_{cg/imu}^b \times \right] \right] \delta \boldsymbol{\alpha}(t) \\ &\quad - \mathbf{T}_b^i(\hat{\mathbf{q}}_i^b(t)) \mathbf{T}_c^b (\delta \mathbf{b}_a(t) + \mathbf{M}_1(\mathbf{a}_{ng,m}^c(t)) \delta \boldsymbol{\gamma}_a(t) + \mathbf{M}_2(\mathbf{a}_{ng,m}^c(t)) \delta \mathbf{s}_a(t) + \boldsymbol{\eta}_a(t)). \end{aligned} \quad (49)$$

VI. Sensor Modeling

The EKF is a model-based algorithm requiring models of the sensors. In general, there are two classes of sensor models required for high-fidelity analysis of the integrated guidance, navigation, and control (IGN&C) system. These are (1) high-fidelity models to support the high-fidelity analysis in simulation, and (2) sensor models for the navigation algorithm. This paper is concerned only with sensor models for the navigation algorithm. To that end, a navigation sensor model includes the following:

1. A mathematical model represented by a nonlinear equation(s) as a function of the states of the system (such as position, velocity, attitude, map tie, etc.).
2. A measurement mapping matrix comprised of the partial derivatives (of the model above) evaluated at the most recent state estimate.
3. An error model comprised of random noise and systematic errors, including representative values of the uncertainty in the various error sources.

The sensors currently under consideration are:

1. Altimeter
2. Velocimeter
3. Star camera
4. Terrain relative navigation (TRN)
5. Hazard relative navigation (HRN)

The models presented here represent the class of sensors that might be used to support precision EDL. Not all sensors will likely be used in all phases of EDL. For example, for the Morpheus project, the star camera and terrain camera (TRN) were not employed at any time. Additionally, the models used in the EKF are dependent on the actual sensors selected. For example, there are various possibilities for the landing altimeter (first return, slant range, etc.). This paper presents an illustrative set of models.

A. Altimeter

The altimeter is modeled as

$$h_{sph,k} = r_{alt,k}^i - r_{grd,k} + b_{alt} + \eta_{alt,k}, \quad (50)$$

where $r_{alt,k}^i = \|\mathbf{r}_{imu,k}^i + \mathbf{T}_b^i(\hat{\mathbf{q}}_{i,k}^b)\mathbf{r}_{alt/imu}^b\|$, $r_{grd,k}$ is the distance of the sub-spacecraft surface point from the center of the planet, $\mathbf{r}_{alt/imu}^b$ is assumed known, b_{alt} is the altimeter sensor bias, and $\eta_{alt,k}$ is a zero-mean, white noise sequence with $E\{\eta_{alt,k}\} = 0, \forall t_k$ and $E\{\eta_{alt,k}^2\} = R_{alt,k}\delta_{k,j} \in \mathbb{R}$, $R_{alt,k} > 0, \forall t_k$. We often model the altimeter bias, b_{alt} , as a random constant with

$$\dot{b}_{alt} = 0, \quad (51)$$

where $E\{b_{alt}\} = 0$ and $E\{b_{alt}^2\} = P_{b_{alt}} > 0 \in \mathbb{R}$. As with other biases (e.g., IMU), we might employ a random constants model driven by zero-mean, white noise processes with given process noise strength or as exponentially correlated random variables with given time constants and process noise strengths (see Gelb⁴). There are various ways to model $r_{grd,k}$ in the EKF. For example, we might model the planet as a sphere and $r_{grd,k}$ would be the spherical radius of the planet plus a term to account for the local surface topography (possibly the mean height of the planet surface above the spherical radius) plus an additional time-varying term to account for deviations from the mean height and for small to medium surface features (e.g., rocks and small craters).

The associated estimate of the altimeter measurement is computed via

$$\hat{h}_{sph,k} = \hat{r}_{alt,k}^i - \hat{r}_{grd,k} + \hat{b}_{alt}, \quad (52)$$

where $\hat{r}_{alt,k}^i = \|\hat{\mathbf{r}}_{imu,k}^i + \mathbf{T}_b^i(\hat{\mathbf{q}}_{i,k}^b)\mathbf{r}_{alt/imu}^b\|$, $\hat{r}_{grd,k}$ is an estimate of the distance of the sub-spacecraft surface point from the center of the planet, and \hat{b}_{alt} is the altimeter sensor bias estimate. Defining $\delta h_{sph,k} := h_{sph,k} - \hat{h}_{sph,k}$, it follows that to first-order we have

$$\delta h_{sph,k} = \frac{\hat{\mathbf{r}}_{alt,k}^{iT}}{\hat{r}_{alt,k}^i} \left(\delta \mathbf{r}_{imu,k}^i - \mathbf{T}_b^i(\hat{\mathbf{q}}_{i,k}^b)[\mathbf{r}_{alt/imu}^b \times] \delta \boldsymbol{\alpha}_k \right) - \delta r_{grd,k} + \delta b_{alt} + \eta_{alt,k}, \quad (53)$$

where $\hat{\mathbf{r}}_{alt,k}^i = \hat{\mathbf{r}}_{imu,k}^i + \mathbf{T}_b^i(\hat{\mathbf{q}}_{i,k}^b)\mathbf{r}_{alt/imu}^b$, $\delta b_{alt} := b_{alt} - \hat{b}_{alt}$ and $\delta r_{grd,k} := r_{grd,k} - \hat{r}_{grd,k}$.

B. Velocimeter

The velocimeter is modeled via

$$\mathbf{v}_{rel,k}^v = \mathbf{T}_b^v \mathbf{T}_b^i(\hat{\mathbf{q}}_{i,k}^b) \left[\mathbf{v}_{vel,k}^i - \boldsymbol{\omega}_{f/i,k}^i \times \mathbf{r}_{vel,k}^i \right] + \mathbf{b}_{vel} + \boldsymbol{\eta}_{vel,k} \quad (54)$$

where

$$\mathbf{r}_{vel,k}^i = \mathbf{r}_{imu,k}^i + \mathbf{T}_b^i(\bar{\mathbf{q}}_{i,k}^b) \mathbf{r}_{vel/imu}^b, \quad \mathbf{v}_{vel,k}^i = \mathbf{v}_{imu,k}^i + \mathbf{T}_b^i(\bar{\mathbf{q}}_{i,k}^b) \left(\boldsymbol{\omega}_{b/i,k}^b \times \mathbf{r}_{vel/imu}^b \right), \quad (55)$$

$\mathbf{r}_{vel/imu}^b$ is assumed known, $\boldsymbol{\omega}_{f/i,k}^i = \mathbf{T}_{f,k}^i \boldsymbol{\omega}_{f/i}^f$ is the angular velocity of the planet-centered, planet-fixed reference frame of the planet with respect to the inertial reference frame represented in the inertial reference frame and $\boldsymbol{\omega}_{f/i}^f$ is the constant known rotation vector of the planet about the spin axis, $\mathbf{r}_{vel,k}^i$ and $\mathbf{v}_{vel,k}^i$ are the position and velocity of the velocimeter in the inertial reference frame, respectively, \mathbf{b}_{vel} is the velocimeter bias in the velocimeter reference frame, and $\boldsymbol{\eta}_{vel,k}$ is a zero-mean, Gaussian, white-noise sequence with $\mathbb{E}\{\boldsymbol{\eta}_{vel,k}\} = 0, \forall t_k$ and $\mathbb{E}\{\boldsymbol{\eta}_{vel,k} \boldsymbol{\eta}_{vel,k}^T\} = \mathbf{R}_{vel,k} \delta_{k,j} \in \mathbb{R}^{3 \times 3}, \mathbf{R}_{vel,k} > 0, \forall t_k$. We often model the velocimeter bias, \mathbf{b}_{vel} , as a random constant with

$$\dot{\mathbf{b}}_{vel} = 0, \quad (56)$$

where $\mathbb{E}\{\mathbf{b}_{vel}\} = 0$ and $\mathbb{E}\{\mathbf{b}_{vel} \mathbf{b}_{vel}^T\} = \mathbf{P}_{b_{vel}} > 0 \in \mathbb{R}^{3 \times 3}$. The estimated relative velocity measurement is

$$\hat{\mathbf{v}}_{rel,k}^v = \mathbf{T}_b^v \mathbf{T}_b^i(\hat{\mathbf{q}}_{i,k}^b) \left(\hat{\mathbf{v}}_{vel,k}^i - \boldsymbol{\omega}_{f/i,k}^i \times \hat{\mathbf{r}}_{vel,k}^i \right) + \hat{\mathbf{b}}_{vel}, \quad (57)$$

where

$$\hat{\mathbf{r}}_{vel,k}^i = \hat{\mathbf{r}}_{imu,k}^i + \mathbf{T}_b^i(\hat{\mathbf{q}}_{i,k}^b) \mathbf{r}_{vel/imu}^b, \quad \hat{\mathbf{v}}_{vel,k}^i = \hat{\mathbf{v}}_{imu,k}^i + \mathbf{T}_b^i(\hat{\mathbf{q}}_{i,k}^b) \left(\hat{\boldsymbol{\omega}}_{b/i,k}^b \times \mathbf{r}_{vel/imu}^b \right), \quad (58)$$

With $\delta \mathbf{v}_{rel,k}^v := \mathbf{v}_{rel,k}^v - \hat{\mathbf{v}}_{rel,k}^v$, $\delta \mathbf{b}_{vel} := \mathbf{b}_{vel} - \hat{\mathbf{b}}_{vel}$, and Eq. (40), it follows that to first-order we have

$$\begin{aligned} \delta \mathbf{v}_{rel,k}^v = & -\mathbf{T}_b^v \mathbf{T}_b^i(\hat{\mathbf{q}}_{i,k}^b) \left(\left[\boldsymbol{\omega}_{f/i,k}^i \times \right] \delta \mathbf{r}_{imu,k}^i - \delta \mathbf{v}_{imu,k}^i \right) - \mathbf{T}_b^v \left[\left(\boldsymbol{\omega}_{b/i,m_k}^b \times \mathbf{r}_{vel/imu}^b \right) \times \right] \\ & + \mathbf{T}_b^i(\hat{\mathbf{q}}_{i,k}^b) \left[\boldsymbol{\omega}_{f/i,k}^i \times \right] \mathbf{T}_b^i(\hat{\mathbf{q}}_{i,k}^b) \left[\mathbf{r}_{vel/imu}^b \times \right] + \left[\mathbf{T}_b^i(\hat{\mathbf{q}}_{i,k}^b) \left(\hat{\mathbf{v}}_{vel}^i - \boldsymbol{\omega}_{f/i,k}^i \times \hat{\mathbf{r}}_{vel}^i \right) \times \right] \delta \boldsymbol{\alpha} \\ & + \mathbf{T}_b^v \left[\mathbf{r}_{vel/imu}^b \times \right] \mathbf{T}_c^b \left(\delta \mathbf{b}_g + \mathbf{N}_1(\boldsymbol{\omega}_{b/i,m_k}^c) \boldsymbol{\gamma}_g + \mathbf{N}_2(\boldsymbol{\omega}_{b/i,m_k}^c) \delta \mathbf{s}_g + \boldsymbol{\eta}_g \right) + \delta \mathbf{b}_{vel} + \boldsymbol{\eta}_{vel}, \end{aligned} \quad (59)$$

where $\boldsymbol{\omega}_{b/i,m_k}^c$ is the measured angular velocity at t_k and $\boldsymbol{\omega}_{b/i,m_k}^b = \mathbf{T}_c^b \boldsymbol{\omega}_{b/i,m_k}^c$.

C. Star Camera

The star camera is modeled via

$$\bar{\mathbf{q}}_{sr,k}^{sc} = \bar{\mathbf{q}}_{\eta,k} \otimes \bar{\mathbf{q}}_b^{sc} \otimes \bar{\mathbf{q}}_{i,k}^b \otimes \bar{\mathbf{q}}_{sr}^i, \quad (60)$$

where $\bar{\mathbf{q}}_b^{sc}$ and $\bar{\mathbf{q}}_{sr}^i$ are known transformations, and $\bar{\mathbf{q}}_{\eta,k}$ is the bias-noise quaternion given by

$$\bar{\mathbf{q}}_{\eta,k} = \begin{bmatrix} \sin\left(\frac{\theta_k}{2}\right) \frac{\boldsymbol{\theta}_k}{\theta_k} \\ \cos\left(\frac{\theta_k}{2}\right) \end{bmatrix},$$

where $\boldsymbol{\theta}_k = \mathbf{b}_{sc} + \boldsymbol{\eta}_{sc,k}$, $\theta_k = \|\boldsymbol{\theta}_k\|$, and $\boldsymbol{\eta}_{sc,k}$ is a zero-mean, Gaussian, white-noise sequence with $\mathbb{E}\{\boldsymbol{\eta}_{sc,k}\} = 0, \forall t_k$ and $\mathbb{E}\{\boldsymbol{\eta}_{sc,k} \boldsymbol{\eta}_{sc,k}^T\} = \mathbf{R}_{sc,k} \delta_{k,j} \in \mathbb{R}^{3 \times 3}, \mathbf{R}_{sc,k} > 0, \forall t_k$. We often model the star camera bias, \mathbf{b}_{sc} , as a random constant with

$$\dot{\mathbf{b}}_{sc} = 0, \quad (61)$$

where $\mathbb{E}\{\mathbf{b}_{sc}\} = 0$ and $\mathbb{E}\{\mathbf{b}_{sc} \mathbf{b}_{sc}^T\} = \mathbf{P}_{b_{sc}} > 0 \in \mathbb{R}^{3 \times 3}$.

The estimated star camera quaternion is

$$\hat{\mathbf{q}}_{sr,k}^{sc} = \hat{\mathbf{q}}_{\eta,k} \otimes \bar{\mathbf{q}}_b^{sc} \otimes \hat{\mathbf{q}}_{i,k}^b \otimes \bar{\mathbf{q}}_{sr}^i. \quad (62)$$

where $\hat{\boldsymbol{\theta}}_k = \hat{\mathbf{b}}_{sc}$. With the definitions $\delta \bar{\mathbf{q}}_{sr,k}^{sc} := \bar{\mathbf{q}}_{sr,k}^{sc} \otimes \hat{\mathbf{q}}_{sr,k}^{sc-1}$, $\delta \bar{\mathbf{q}}_{i,k}^b := \bar{\mathbf{q}}_{i,k}^b \otimes \hat{\mathbf{q}}_{i,k}^{b-1}$, and $\delta \bar{\mathbf{q}}_{\eta,k} := \bar{\mathbf{q}}_{\eta,k} \otimes \hat{\mathbf{q}}_{\eta,k}^{-1}$, and from Eq. (60) and Eq. (62), it follows that to first-order we have

$$\delta \bar{\mathbf{q}}_{sr,k}^{sc} = \begin{bmatrix} \delta \mathbf{q}_{\eta,k} + \mathbf{T}(\hat{\mathbf{q}}_{\eta,k}) \mathbf{T}_b^{sc} \delta \mathbf{q}_{i,k}^b \\ 1 \end{bmatrix}. \quad (63)$$

Assuming small angles and with the definition $\delta \mathbf{b}_{sc} := \mathbf{b}_{sc} - \hat{\mathbf{b}}_{sc}$, we have

$$\delta \mathbf{q}_{\eta,k} = \frac{1}{2} \delta \mathbf{b}_{sc} + \frac{1}{2} \boldsymbol{\eta}_{sc,k}. \quad (64)$$

Then with $\delta \mathbf{q}_{i,k}^b \approx \frac{1}{2} \delta \boldsymbol{\alpha}_k$ and $\delta \mathbf{q}_{sr,k}^{sc} \approx \frac{1}{2} \delta \boldsymbol{\psi}_k$, substituting Eq. (64) into Eq. (63) yields

$$\delta \boldsymbol{\psi} = \mathbf{T}(\hat{\mathbf{q}}_{\eta,k}) \mathbf{T}_b^{sc} \delta \boldsymbol{\alpha}_k + \delta \mathbf{b}_{sc} + \boldsymbol{\eta}_{sc,k}. \quad (65)$$

D. Hazard Relative Navigation

HRN tracking is comprised of an hazard detection and avoidance (HDA) phase to create a feature map followed by a navigation phase in which HRN measurements are processed. The feature map specifies the locations of the key features relative to the corrected intended landing site (ILS) where the corrected ILS incorporates that latest estimates of the map-tie errors. As the descent progresses and when the resolution dictates, a new feature map is generated—this mapping process is invisible to the navigation algorithm except that HRN measurements are not provided for short periods during the mapping—and the navigation phase continues once the HRN measurements are again provided. During each navigation phase, we assume that only one feature is tracked.

The first step in the HDA process is for navigation to provide to the HDA algorithm the current inertial state estimate for the purpose of locating the HRN sensor with respect to the corrected ILS. The location of the HRN sensor relative to the corrected ILS is given in the planet-surface, planet-fixed reference frame and the location of the feature relative to the HRN sensor is given in the HRN sensor reference frame.

The corrected ILS, represented by $\hat{\mathbf{r}}_{ILS}^f$ in Fig. 6, is given by Eq. (30) where h_{ls} , ϕ_{ls} , and λ_{ls} , respectively, are assumed known *a priori*, and the associated map-tie errors denoted by \hat{h}_Δ , $\hat{\phi}_\Delta$ and $\hat{\lambda}_\Delta$, respectively, are estimated before the HDA process starts (likely during the TRN phase). As depicted in Fig. 6, the estimated location of the HRN sensor relative to the corrected ILS at $t = t_j$ is given by

$$\hat{\mathbf{r}}_{ILS/HRN,j}^s = -\hat{\mathbf{T}}_f^s \left(\mathbf{T}_{i,j}^f \left(\hat{\mathbf{r}}_{IMU,j}^i + \mathbf{T}_b^i(\hat{\mathbf{q}}_{i,j}^b) \mathbf{r}_{HRN/IMU}^b \right) - \hat{\mathbf{r}}_{ILS}^f \right), \quad (66)$$

where

$$\hat{\mathbf{T}}_f^s = \begin{bmatrix} -\sin(\lambda_{ls} + \hat{\lambda}_\Delta) & \cos(\lambda_{ls} + \hat{\lambda}_\Delta) & 0 \\ -\sin(\phi_{ls} + \hat{\phi}_\Delta) \cos(\lambda_{ls} + \hat{\lambda}_\Delta) & -\sin(\phi_{ls} + \hat{\phi}_\Delta) \sin(\lambda_{ls} + \hat{\lambda}_\Delta) & \cos(\phi_{ls} + \hat{\phi}_\Delta) \\ \cos(\phi_{ls} + \hat{\phi}_\Delta) \cos(\lambda_{ls} + \hat{\lambda}_\Delta) & \cos(\phi_{ls} + \hat{\phi}_\Delta) \sin(\lambda_{ls} + \hat{\lambda}_\Delta) & \sin(\phi_{ls} + \hat{\phi}_\Delta) \end{bmatrix},$$

and $\mathbf{r}_{HRN/IMU}^b$ is known. During the mapping phase, we compute the locations of tracking features in the planet-surface, planet-fixed reference frame with the origin at the corrected ILS. Referring to Fig. 7, we obtain the relationship $\hat{\mathbf{r}}_{F/ILS}^s$ as

$$\hat{\mathbf{r}}_{F/ILS}^s = \hat{\mathbf{T}}_f^s \mathbf{T}_{i,m}^f \mathbf{T}_b^i(\hat{\mathbf{q}}_{i,m}^b) \mathbf{T}_h^b \mathbf{y}_{F/HRN,m}^h - \hat{\mathbf{r}}_{ILS/HRN,m}^s, \quad (67)$$

where $t = t_m$, $\hat{\mathbf{q}}_{i,m}^b$ and $\hat{\mathbf{r}}_{ILS/HRN,m}^s$ are propagated from $t = t_j$ to $t = t_m$ using an HRN-based IMU initialized by state estimates from the EKF to compute $\hat{\mathbf{q}}_{i,j}^b$ and $\hat{\mathbf{r}}_{ILS/HRN,j}^s$, respectively, and $\mathbf{y}_{F/HRN,m}^h$ is the HRN measurement where,

$$\mathbf{y}_{F/HRN,m}^h = \mathbf{r}_{F/HRN,m}^h + \mathbf{b}_{hrn} + \boldsymbol{\eta}_{hrn,m} \quad (68)$$

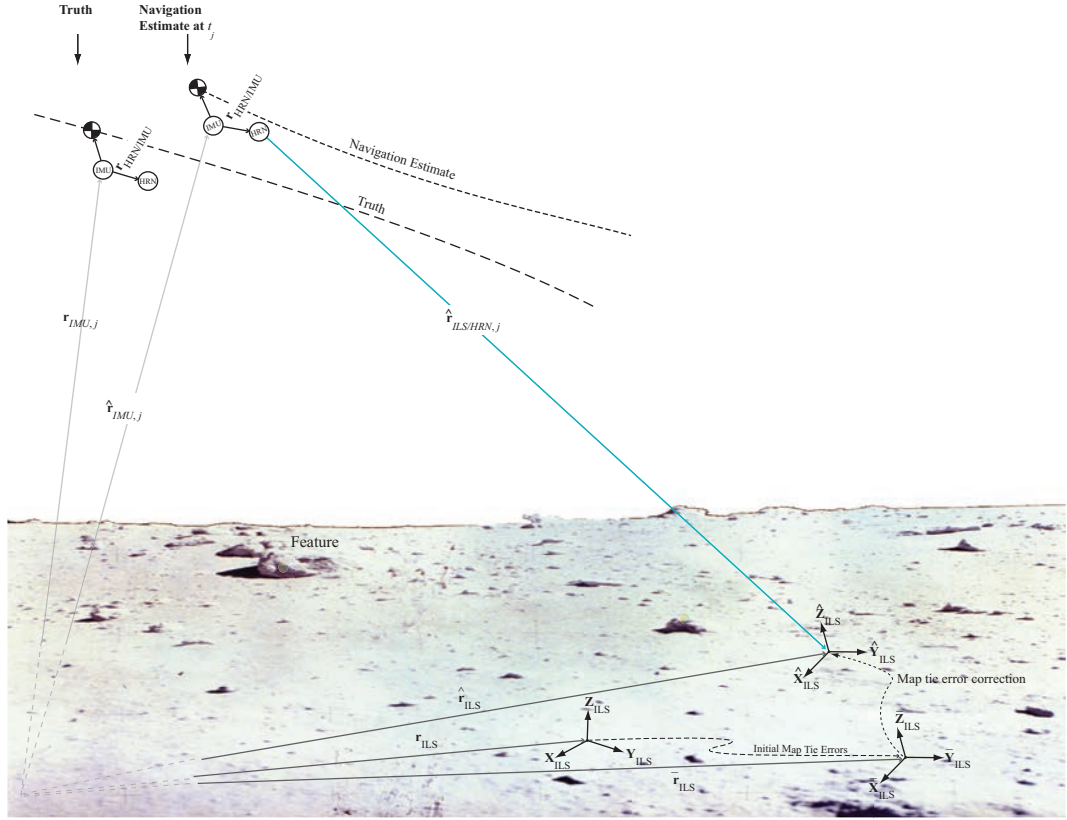


Figure 5. HDA modeling—Step 1—Locating the HRN sensor.

\mathbf{b}_{hrn} is the HRN bias in the HRN reference frame, and $\boldsymbol{\eta}_{hrn,m}$ is a zero-mean, Gaussian, white-noise sequence with $E\{\boldsymbol{\eta}_{hrn,m}\} = 0, \forall t_m$ and $E\{\boldsymbol{\eta}_{hrn,m}\boldsymbol{\eta}_{hrn,j}^T\} = \mathbf{R}_{hrn,m}\delta_{m,j} \in \mathbb{R}^{3 \times 3}, \mathbf{R}_{hrn,m} > 0, \forall t_m$. We often model the HRN bias, \mathbf{b}_{hrn} , as a random constant with

$$\dot{\mathbf{b}}_{hrn} = 0, \quad (69)$$

where $E\{\mathbf{b}_{hrn}\} = 0$ and $E\{\mathbf{b}_{hrn}\mathbf{b}_{hrn}^T\} = \mathbf{P}_{\mathbf{b}_{hrn}} > 0 \in \mathbb{R}^{3 \times 3}$.

Consider the navigation phase of the HRN depicted in Fig. 8. Using the measurement model in Eq. (76) we can write

$$\hat{\mathbf{y}}_{F/HRN,m}^h = \hat{\mathbf{r}}_{F/HRN,m}^h + \hat{\mathbf{b}}_{hrn}. \quad (70)$$

To compute $\delta\mathbf{y}_{F/HRN,m}^h := \mathbf{y}_{F/HRN,m}^h - \hat{\mathbf{y}}_{F/HRN,m}^h$, consider Fig. 8 to obtain

$$\mathbf{r}_{F/HRN,k}^h = -\mathbf{T}_b^h \left[\mathbf{T}_b^i(\hat{\mathbf{q}}_{i,k}^b) \mathbf{r}_{IMU,k}^i + \mathbf{r}_{HRN/IMU}^b - \mathbf{T}_b^i(\hat{\mathbf{q}}_{i,k}^b) \mathbf{T}_f^i \left(\hat{\mathbf{r}}_{ILS}^f + \hat{\mathbf{T}}_s^f \left(\hat{\mathbf{r}}_{F/ILS}^s - \Delta\mathbf{r}_F^s \right) \right) \right],$$

from which it follows that

$$\hat{\mathbf{r}}_{F/HRN,k}^h = -\mathbf{T}_b^h \left[\mathbf{T}_b^i(\hat{\mathbf{q}}_{i,k}^b) \hat{\mathbf{r}}_{IMU,k}^i + \mathbf{r}_{HRN/IMU}^b - \mathbf{T}_b^i(\hat{\mathbf{q}}_{i,k}^b) \mathbf{T}_f^i(t_k) \left(\hat{\mathbf{r}}_{ILS}^f + \hat{\mathbf{T}}_s^f \left(\hat{\mathbf{r}}_{F_i/ILS}^s - \hat{\Delta}\mathbf{r}_{F,k}^s \right) \right) \right].$$

Defining $\delta\mathbf{b}_{hrn,k} := \mathbf{b}_{hrn} - \hat{\mathbf{b}}_{hrn,k}$, $\delta\Delta\mathbf{r}_{F,k}^s := \Delta\mathbf{r}_F^s - \hat{\Delta}\mathbf{r}_{F,k}^s$ and $\delta\mathbf{r}_{F/HRN,k}^h := \mathbf{r}_{F/HRN,k}^h - \hat{\mathbf{r}}_{F/HRN,k}^h$, it follows that

$$\delta\mathbf{y}_{F/HRN,m}^h = \delta\mathbf{r}_{F/HRN,k}^h + \delta\mathbf{b}_{hrn,k} + \boldsymbol{\eta}_{hrn,k}, \quad (71)$$

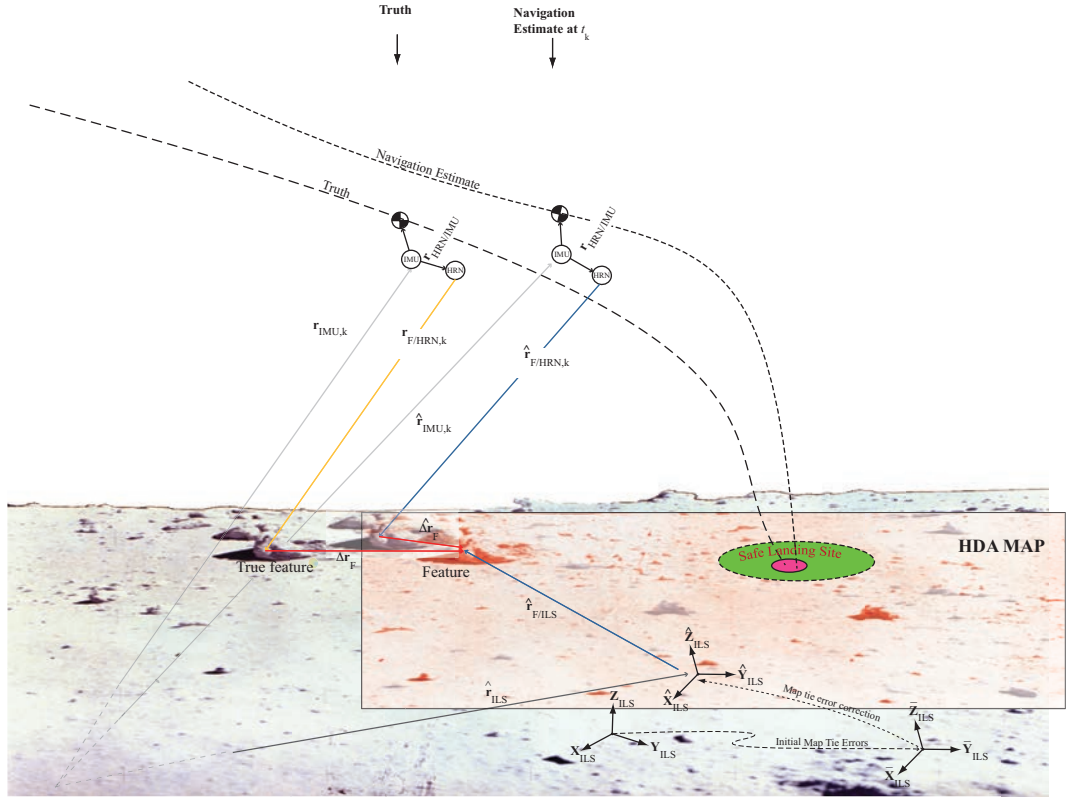


Figure 7. HRN navigation phase.

where the parameter vectors for the accelerometer, gyro, map tie errors, and sensor biases are defined as

$$\mathbf{p}_a = \begin{pmatrix} \mathbf{b}_a \\ \mathbf{s}_a \\ \gamma_a \end{pmatrix}, \quad \mathbf{p}_g = \begin{pmatrix} \mathbf{b}_g \\ \mathbf{s}_g \\ \gamma_g \end{pmatrix}, \quad \mathbf{p}_m = \begin{pmatrix} h_\Delta \\ \phi_\Delta \\ \lambda_\Delta \end{pmatrix}, \quad \text{and} \quad \mathbf{p}_s = \begin{pmatrix} b_{alt} \\ \mathbf{b}_{vel} \\ \mathbf{b}_{sc} \\ \mathbf{b}_{trn} \end{pmatrix}.$$

Therefore, $\mathbf{p}_a \in \mathbb{R}^{12}$, $\mathbf{p}_g \in \mathbb{R}^{12}$, $\mathbf{p}_m \in \mathbb{R}^3$, and $\mathbf{p}_s \in \mathbb{R}^{10}$.

The TRN and HRN will not be used for navigation at the same time. This enables the reuse of several key state variables. During the TRN phase we expect that the map-tie errors will be reduced, hence we will estimate $\mathbf{p}_m = (h_\Delta, \phi_\Delta, \lambda_\Delta)$. Once the HRN phase begins, we stop updating the corrected ILS to ensure that guidance and targeting are not chasing a moving landing site. However, during HRN we will need to estimate $\Delta \mathbf{r}_F^s \in \mathbb{R}^3$, so we reset $\mathbf{p}_m \rightarrow \Delta \mathbf{r}_F^s$. We can also reset $\mathbf{b}_{trn} \rightarrow \mathbf{b}_{hrn}$.

One caveat to the state estimate update given by Eq. (12) is that of the attitude update. All of the preceding developments have focused on the attitude error in the form of small angles, denoted by $\delta \boldsymbol{\alpha}_k$. If the portion of $\hat{\mathbf{x}}_k^+$ pertaining to the attitude is given by $\delta \hat{\boldsymbol{\alpha}}_k^+$, then the quaternion update is

$$\hat{\mathbf{q}}_{i,k}^{b+} = \begin{bmatrix} \frac{1}{2} \delta \hat{\boldsymbol{\alpha}}_k^+ \\ 1 \end{bmatrix} \otimes \hat{\mathbf{q}}_{i,k}^{b-}, \quad (74)$$

where $\hat{\mathbf{q}}_{i,k}^{b-}$ is the *a priori* estimate of the quaternion and $\delta \hat{\boldsymbol{\alpha}}_k^- = 0$. The quaternion in Eq. (82) satisfies the unity norm constraint only to first order, and so to ensure that the quaternion remains unity norm a normalization is performed.

VIII. Acknowledgements

References

- ¹Rudolph E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Transactions of the ASME – Journal of Basic Engineering*, Vol. 82, 1960, pp. 35–45.
- ²Archinal, B.A., *et al.*, "Report of the IAU/IAG/COSPAR Working Group on Cartographic Coordinates and Rotational Elements of the Planets and Satellites: 2009" *Celestial Mechanics and Dynamical Astronomy*, Vol. 109, No. 2, 2011.
- ³"Planetary Data System Standards Reference," JPL D-7669, Pt. 2, Ver. 3.7, March 2006.
- ⁴Arthur Gelb, *Applied Optimal Estimation*, The MIT Press, Cambridge, MA, 1974.
- ⁵Peter S. Maybeck, *Stochastic Models, Estimation, and Control (Volume 1)*, Academic Press, 1979.
- ⁶Byron D. Tapley, Bob E. Schutz, and George H. Born, *Statistical Orbit Determination*, Elsevier Academic Press, New York, NY, 2004.
- ⁷Pedro R. Escobal, *Methods of Orbit Determination*, Krieger Publishing Company, Malabar, FL, 1965.
- ⁸Wolfgang Torge, *Geodesy, 3rd Edition*, Walter de Gruyter, Berlin, 2001.
- ⁹Peter Lancaster, *Theory of Matrices*, Academic Press, New York, NY, 1969.
- ¹⁰Marshall H. Kaplan, *Modern Spacecraft Dynamics & Control*, John Wiley & Sons, New York, NY, 1976.
- ¹¹Kenneth R. Britting, *Inertial Navigation Systems Analysis*, John Wiley & Sons, New York, NY, 1971.
- ¹²Timothy P. Crain and Robert H. Bishop, "Mars Entry Navigation: Atmospheric Interface through Parachute Deploy," *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, 2002.
- ¹³Ralph B. Roncoli, "Lunar Constants and Models Document," *JPL Technical Document D-32296*, 2005.
- ¹⁴Acton, C., Navigation and Ancillary Information Facility, available online at: <http://naif.jpl.nasa.gov/naif/>.